AD 676974

MEMORANDUM
RM-5707-PR
SEPTEMBER 1968

TWO-PARAMETER EXPONENTIAL AND RATIONAL FUNCTIONS FOR LEAST-SQUARE APPROXIMATIONS

M. I. Liechenstein

This research is supported by the United States Air Force under Project RAND—Contract No. F44620-67-C-0045—monitored by the Directorate of Operational Requirements and Development Plans, Deputy Chief of Staff, Research and Development, Hq USAF. Views or conclusions contained in this study should not be interpreted as representing the official opinion or policy of the United States Air Force.

DISTRIBUTION STATEMENT
This document has been approved for public release and sale; its distribution is unlimited.

The RAND Corporation
1700 MAIN ST. • SANTA MONICA • CALIFORNIA • 90406

Published by The RAND Corporation

PREFACE

As a contribution to The RAND Corporation's studies in system analysis and synthesis, a scheme is delineated in this report for approximating prescribed system characteristics by sums of exponentials or by rational functions. The analytical results and computational aids which are derived are applicable to a broad class of system optimizations, such as those found in radar filter design; numerical examples are provided to illustrate their application to practical and important design problems. The contents of this Memorandum should be of interest to the Air Force Systems Command, as well as to others concerned with numerical methods of system design and signal analysis.

SUMMARY

In many system design problems, the representation of certain system attributes must be in terms of exponentially damped sinusoids or of rational functions in order to be physically meaningful. In this Memorandum, two sets of orthonormal elements are derived which should be useful for such approximation problems. One set constitutes a basis for exponential approximation and the other a basis for rational function approximation.

The closure properties of the two-parameter exponential and rational functions are examined first. Expressions are then presented for efficiently determining the orthonormal elements of each basis. Important characteristics of the sets are also discussed, and special relations among the coefficients generating the bases are deduced.

Once the general relations for the orthonormal elements are developed, the two sets are applied to typical approximation problems encountered in signal processing and system design. The computations involved in the solution of these problems are illustrated in the final portion of the study. The computer programs used for the sample problems are described in the appendices and should be helpful in similar design situations.

ACKNOWLEDGMENTS

It is a pleasure to acknowledge the meticulous reviews of Robert E. Kalaba and Stanley P. Azen of The RAND Corporation.

CONTENTS

LIST OF FIGURES

LIST OF TABLES

I. INTRODUCTION

To ensure physical realizability in the synthesis of a system, it is frequently necessary to represent certain of the system's attributes by approximants other than algebraic polynomials. Such is the case, for instance, in the determination of optimum filters for smoothing and prediction where it is convenient to approximate empirical or analytical spectra of random processes by readily factorized rational functions.† The utility of exponential functions is well known in the design of linear networks for a prescribed transient response. Similarly, sums of exponentials often afford the most suitable approximations to cross-correlation measurements, to radioactive decay and gas absorption data, to mass spectrographs, and to ultracentrifuge analysis curves.

In the present treatment of rational and exponential function representations, two orthonormal bases for exponential and rational function approximations are derived. The bases consist of two-parameter elements which provide more efficient minimum mean-square-error approximation than two corresponding one-parameter sets investigated previously.(1) After the closure properties of the two orthonormal bases are examined in detail, new expressions are developed for efficiently generating the individual orthonormal elements. Several relations are then deduced which connect important properties of each basis. Useful identities and numerical techniques involving the basis coefficients are derived

† The term "rational function" denotes a ratio of algebraic polynomials in which the denominator polynomial is not generally a constant.

which obviate storage of either the form or selected values of the orthonormal approximants. Finally, several numerical examples are provided to illustrate both algorithms for exponential and rational function approximation.

II. SCOPE OF INVESTIGATION

In a previous investigation, the linearly independent functions

$$\hat{\varphi}_n(t) = \begin{cases} e^{-n\sigma t} & 0 < t < \infty \\ 0 & t \le 0 \end{cases} \tag{1}$$

$$n = 1,2,\ldots;\ \mathrm{Im}(\sigma) = 0;\ \mathrm{Re}(\sigma) > 0$$

and

$$\hat{\psi}_n(\omega) = \frac{1}{n\sigma + j\omega} \quad -\infty < \omega < \infty \tag{2}$$

$$n = 1,2,\ldots;\ \mathrm{Im}(\sigma) = 0;\ \mathrm{Re}(\sigma) > 0$$

were examined in detail.† The more general sets comprised of

$$\varphi_n(t) = \begin{cases} e^{-n(\sigma - j\beta)t} & 0 < t < \infty \\ 0 & t \le 0 \end{cases} \tag{3}$$

$$n = 1,2,\ldots;\ \mathrm{Im}(\sigma) = \mathrm{Im}(\beta) = 0;\ \mathrm{Re}(\sigma) > 0$$

and

$$\psi_n(\omega) = \frac{1}{n\sigma + j(\omega - n\beta)} \quad -\infty < \omega < \infty \tag{4}$$

$$n = 1,2,\ldots;\ \mathrm{Im}(\sigma) = \mathrm{Im}(\beta) = 0;\ \mathrm{Re}(\sigma) > 0$$

will be treated in this Memorandum. The primary motivation for such a study is that for a fixed number of orthonormal elements, an expansion of a prescribed function in terms of $\varphi_n(t)$ or $\psi_n(\omega)$ provides a minimum mean-square-error less than or equal to that for $\hat{\varphi}_n(t)$ or $\hat{\psi}_n(\omega)$, respectively.

---

† See Tables 1-7 of Ref. 1.

In the remaining sections, the results obtained for the one-parameter sets, Eqs. (1) and (2), are extended to $\{\varphi_n(t)\}$ and $\{\psi_n(\omega)\}$, and numerical schemes are derived for simplifying least-square representations involving $\{\varphi_n(t)\}$ and $\{\psi_n(\omega)\}$. In so doing, the following topics are considered:

- Closure of $\{\varphi_n(t)\}$ and $\{\psi_n(\omega)\}$
- Determination of the orthonormalized sets†

$$X_m(t) = \sum_{n=1}^{m} \gamma_{mn}\, \varphi_n(t), \qquad Y_m(t) = \frac{1}{\sqrt{2}}\,[X_m(t) + X_m^*(t)]$$

and

$$U_m(\omega) = \sum_{n=1}^{m} \lambda_{mn}\, \psi_n(\omega), \qquad V_m(\omega) = \frac{1}{\sqrt{2}}\,[U_m(\omega) + U_m^*(\omega)]$$

- Properties of X_m, Y_m, U_m, V_m, γ_{mn}, and λ_{mn}
- Selection of σ and β
- Computational aspects of approximants

$$g(t) \approx \sum_{m=1}^{M} a_m X_m(t) \quad \text{and} \quad h(\omega) \approx \sum_{m=1}^{M} b_m U_m(\omega)$$

---

†The symbol * denotes complex conjugate.

III. CLOSURE

In order to approximate prescribed functions arbitrarily closely by linear combinations of the elements $\varphi_n(t)$ and $\psi_n(\omega)$, it is necessary to verify the closure of the sets $\{\varphi_n(t)\}$ and $\{\psi_n(\omega)\}$. Since the functions $\varphi_n(t)$ and $\psi_n(\omega)$ are related through a linear operation, the Fourier transform, it will be possible to demonstrate closure of $\{\psi_n(\omega)\}$ in the space $L^2(-\infty,\infty)$ once closure of $\{\varphi_n(t)\}$ in $L^2(0,\infty)$ is shown.

An assemblage of functions $\{\varphi_n(t)\}$ of integrable square† is <u>closed</u> over (a,b) if the integral

$$\int_a^b y(t)\ \varphi_n^*(t)\ dt = 0 \qquad n = 1,2,... \tag{5}$$

implies that $y(t) \subset L^2(a,b)$ vanishes everywhere in (a,b) except perhaps over a set of measure zero. The closure property derives its significance from a theorem that states that a set of functions is closed if and only if it is complete.(2) Moreover, if a set $\{\varphi_n(t)\}$ is <u>complete</u>, then for any function $y(t) \subset L^2(a,b)$ and any positive ϵ, there is a sum function

$$\hat{y}(t) = \sum_{n=1}^{N} a_n\ \varphi_n(t) \tag{6}$$

such that

$$\int_a^b |y(t) - \hat{y}(t)|^2\ dt < \epsilon \tag{7}$$

†This property is symbolized by $\varphi_n(t) \subset L^2(a,b)$.

Equations (5) and (6) represent a special case of the general situation in which a closed finite or infinite system of elements $\{y_n(x)\}$ in a Hilbert space Y permits arbitrarily close approximation (in the L^2 norm) to every element $y(x) \subset Y$ by a finite linear combination of the $y_n(x)$. To establish this property for the elements $\varphi_n(t)$ in Eq. (3), Szasz's theorem[†] can be invoked:

> A necessary and sufficient condition for closure $L^2(0,1)$ of the functions x^{λ_n}, $\mathrm{Re}(\lambda_n) > -\frac{1}{2}$, is divergence of

$$\sum_{n=1}^{\infty} \frac{1 + 2\mathrm{Re}[\lambda_n]}{1 + |\lambda_n|^2} \tag{8}$$

From the previous definition of closure, if $\{x^{\lambda_n}\}$ is not closed in $L^2(0,1)$, a function $y(x) \subset L^2(0,1)$ exists which is orthogonal to every element x^{λ_n}; that is

$$0 < \int_0^1 |y(x)|^2 \, dx < \infty \tag{9}$$

and

$$\int_0^1 y(x) \, x^{\lambda_n^*} \, dx = 0 \qquad n = 1,2,\ldots \tag{10}$$

Conversely, if the set $\{x^{\lambda_n}\}$ is closed, a function $y(x) \subset L^2(0,1)$, which is not identically zero, does not exist so that Eqs. (9) and (10) are satisfied. Accordingly, with the change of variable $x = e^t$ in the set $\{x^{\lambda_n}\}$, the conditions given in Eqs. (9) and (10) become

---

[†] See Ref. 2, pp. 32-36.

$$0 < \int_{-\infty}^{0} |y(e^t)|^2 \, e^t \, dt < \infty \tag{9'}$$

and

$$\int_{-\infty}^{0} y(e^t) \, e^{t(1+\lambda_n^*)} \, dt = 0 \qquad n = 1,2,\ldots \tag{10'}$$

With the further transformation $y(e^t)\, e^{t/2} = \bar{y}(t)$, the last two relations give

$$0 < \int_{-\infty}^{0} |\bar{y}(t)|^2 \, dt < \infty \tag{9''}$$

and

$$\int_{-\infty}^{0} \bar{y}(t) \, e^{t(\frac{1}{2}+\lambda_n^*)} \, dt = 0 \qquad n = 1,2,\ldots \tag{10''}$$

Thus, closure of the functions x^{λ_n} in the space $\bar{y}(x) \subset L^2(0,1)$ is equivalent to existence of a function $\bar{y}(t)$ satisfying Eqs. (9″) and (10″). This in turn is equivalent to the closure of the functions $e^{t(\frac{1}{2}+\lambda_n)}$ in $L^2(-\infty,0)$, or $e^{-t(\frac{1}{2}+\lambda_n)}$ in $L^2(0,\infty)$.

In the application of interest, where $\varphi_n(t) = e^{-n(\sigma-j\beta)t}$, λ_n satisfies the equation

$$\begin{gathered} \lambda_n = (-\tfrac{1}{2} + n\sigma) - jn\beta \\ n = 1,2,\ldots; \quad \mathrm{Im}(\sigma) = \mathrm{Im}(\beta) = 0; \quad \mathrm{Re}(\sigma) > 0 \end{gathered} \tag{11}$$

Consequently,

$$\mathrm{Re}[\lambda_n] = -\tfrac{1}{2} + n\sigma > -\tfrac{1}{2} \qquad n = 1,2,\ldots; \ \mathrm{Re}(\sigma) > 0, \ \mathrm{Im}(\sigma) = 0 \tag{12}$$

as required by Szasz's theorem. Substituting this value of λ_n in Eq. (8) yields

$$\sum_{n=1}^{\infty} \frac{1 + 2(-\frac{1}{2} + n\sigma)}{1 + (-\frac{1}{2} + n\sigma)^2 + (n\beta)^2} = \sum_{n=1}^{\infty} \frac{2n\sigma}{n^2(\sigma^2+\beta^2) - n\sigma + 5/4} \qquad (13)$$

According to Szasz's theorem, closure of $\{\varphi_n(t) = e^{-n(\sigma - j\beta)t}\}$ in the space $\bar{y}(t) \subset L^2(0,\infty)$ rests on divergence of this sum.

Application of the integral test[3] to the sum in Eq. (13) indicates that divergence of the infinite series depends on divergence of the integral

$$\int_1^{\infty} \frac{2\sigma\xi \, d\xi}{1 + (-\frac{1}{2}+\sigma\xi)^2 + (\beta\xi)^2} \qquad (14)$$

The resultant integration is

$$\frac{2\sigma^2}{(\sigma^2+\beta^2)\sqrt{4\sigma^2+5\beta^2}} \tan^{-1}\left[\frac{2(\sigma^2+\beta^2)\,\xi - \sigma}{\sqrt{4\sigma^2 + 5\beta^2}}\right] \Bigg|_1^{\infty}$$

$$+ \frac{\sigma}{(\sigma^2+\beta^2)} \log\left[(\sigma^2+\beta^2)\,\xi^2 - \sigma\xi + 5/4\right] \Bigg|_1^{\infty} \qquad (15a)$$

or

$$\frac{2\sigma^2}{(\sigma^2+\beta^2)(4\sigma^2+5\beta^2)^{\frac{1}{2}}} \left\{ \frac{\pi}{2} - \tan^{-1}\left[\frac{2(\sigma^2+\beta^2)-\sigma}{(4\sigma^2+5\beta^2)^{\frac{1}{2}}}\right]\right\} - \frac{\sigma}{(\sigma^2+\beta^2)} \log\left[\sigma^2+\beta^2-\sigma+5/4\right]$$

$$+ \lim_{\xi\to\infty} \frac{\sigma}{(\sigma^2+\beta^2)} \log\left[(\sigma^2+\beta^2)\,\xi^2 - \sigma\xi + 5/4\right] \to \infty \qquad (15b)$$

Since σ^2 and β^2 are positive and bounded, the last term of Eq. (15b) is unbounded. Consequently, the series of Eq. (13) is divergent, and the closure conditions of Szasz's theorem are satisfied by the set

$\{\varphi_n(t)\}$. It follows, therefore, that $\{\varphi_n(t) = e^{-n(\sigma - j\beta)t}\}$ is also complete in the space of functions $y(t) \subset L^2(0,\infty)$.

The closure of $\{\psi_n(\omega)\}$ in the transform space $D(\omega) \subset L^2(-\infty,\infty)$ is deducible directly from a lemma of Wiener's on invariance of closure.† His lemma states that the closure of a set of functions is preserved in any linear transformation which carries the whole of L^2 into itself and which conserves the integral of the squared modulus of each function. Quantitatively, the lemma states that:

> Given a linear transformation such that to every function $f(x) \subset L^2(a,b)$, there corresponds a $g(y) \subset L^2(c,d)$, if $f_j(x) \rightarrow g_j(x)$ and
>
> i. $c\, f_j(x) \rightarrow c\, g_j(y)$
>
> ii.†† $f_1(x) + f_2(x) \rightarrow g_1(y) + g_2(y)$
>
> iii. $\int_a^b |f_j(x)|^2\, dx = \int_c^d |g_j(y)|^2\, dy \qquad j = 1,2,\ldots$
>
> then the closure properties of a sequence $\{f_j(x)\}$ are the same as those of $\{g_j(y)\}$.

The transformation which carries the set $\{\varphi_n(t)\}$ into the set $\{\psi_n(\omega)\}$ is the Fourier transform, since

$$\mathfrak{F}\{\varphi_n(t)\} \equiv \int_{-\infty}^{\infty} \varphi_n(t)\, e^{-j\omega t}\, dt = \int_0^{\infty} e^{-(n\sigma + j\omega - jn\beta)t}\, dt$$

$$= \left. \frac{-e^{-(n\sigma + j\omega - jn\beta)t}}{n\sigma + j\omega - jn\beta} \right|_0^{\infty} = \frac{1}{n\sigma + j(\omega - n\beta)} \equiv \psi_n(\omega) \qquad (16)^{\dagger\dagger\dagger}$$

---

†See Ref. 2, pp. 28-30.

††The symbol → denotes correspondence.

†††$\mathfrak{F}$ denotes the Fourier transform.

Because the Fourier transform is a linear operator, properties (i) and (ii) of Wiener's lemma are satisfied. Moreover, Parseval's theorem[4] indicates that

$$\int_0^\infty |\varphi_n(t)|^2 \, dt = \frac{1}{2\pi} \int_{-\infty}^\infty |\psi_n(\omega)|^2 \, d\omega = \int_{-\infty}^\infty |\psi_n(2\pi f)|^2 \, df \qquad (17)$$

where $\omega = 2\pi f$. Hence, all the conditions of the closure invariance lemma are fulfilled, and the set $\{\psi_n(2\pi f)\}$ is also closed in $L^2(-\infty,\infty)$.

IV. ORTHONORMALIZATION OF $\{\varphi_n(t)\}$ and $\{\psi_n(\omega)\}$

The closure properties discussed in Section III ensure completeness of $\{\varphi_n(t)\}$ in the space of squared-integrable functions $d(t) \subset L^2(0,\infty)$ and completeness of $\{\psi_n(\omega)\}$ in $D(\omega) \subset L^2(-\infty,\infty)$. Consequently, the theory of Hilbert spaces can be applied to two further items of computational importance: orthonormalization of $\{\varphi_n(t)\}$ and $\{\psi_n(\omega)\}$, and representations of functions $d(t) \subset L^2(0,\infty)$ by sums of elements $\varphi_n(t)$ and $\psi_n(\omega)$.

The task of orthonormalizing the set $\{\varphi_n(t)\}$ (or, analogously, the set $\{\psi_n(\omega)\}$) entails finding coefficients γ_{mn} in the functions $X_m(t)$,

$$X_m(t) \equiv \sum_{n=1}^{m} \gamma_{mn}\, \varphi_n(t) \qquad m = 1,2,\ldots \tag{18}$$

such that

$$\int_0^\infty X_r(t)\, X_s^*(t)\, dt = \delta_{rs} \tag{19}$$†

The existence of these constants γ_{mn} is guaranteed by Theorem IV.1 of Ref. 5:

> Let $\varphi_1, \varphi_2, \ldots$ be a finite or infinite sequence of elements such that any finite number of elements $\varphi_1, \varphi_2, \ldots \varphi_\kappa$ are linearly independent. Then constants
>
> $$\begin{array}{lll} \gamma_{11} & & \\ \gamma_{21} & \gamma_{22} & \\ \gamma_{31} & \gamma_{32} & \gamma_{33} \\ \vdots & & \end{array}$$

† The symbol δ_{rs} is the Kronecker delta: $\delta_{rs} = 1$ for $r = s$; $\delta_{rs} = 0$, otherwise.

can be found such that the elements

$$X_1 = \gamma_{11}\, \varphi_1$$
$$X_2 = \gamma_{21}\, \varphi_1 + \gamma_{22}\, \varphi_2$$
$$X_3 = \gamma_{31}\, \varphi_1 + \gamma_{32}\, \varphi_2 + \gamma_{33}\, \varphi_3$$
$$\vdots$$

are orthonormal.

The proof of this theorem provides an iterative algorithm, the Gram-Schmidt orthogonalization procedure, for actually determining the functions $X_m(t)$. The recursion is given by the following:

$$\begin{aligned}
x_1 &= \varphi_1 && \text{and} \quad X_1 = x_1/\|x_1\| \\
x_2 &= \varphi_2 - (\varphi_2, X_1)\, X_1 && \text{and} \quad X_2 = x_2/\|x_2\| \\
&\vdots && \qquad \vdots \\
x_{m+1} &= \varphi_{m+1} - \sum_{k=1}^{m} (\varphi_{m+1}, X_k)\, X_k && \text{and} \quad X_{m+1} = x_{m+1}/\|x_{m+1}\|
\end{aligned} \tag{20}$$ [†]

Although the functions $X_m(t)$ can be determined by this iterative procedure, Eqs. (20) provide only an implicit expression for the coefficients γ_{mn} generating $\{X_m(t)\}$. An additional drawback to the above scheme is that the recursive evaluation of the basis elements is exceedingly laborious.[††]

[†] The notation $\|f\|$ denotes $\left[\int_0^\infty f(t)\, f^*(t)\, dt\right]^{\frac{1}{2}}$, and (f,g) symbolizes $\int_0^\infty f(t)\, g^*(t)\, dt$. Thus, $\|f\|^2 = (f,f)$.

[††] See Ref. 1, pp. 12-14.

To circumvent these difficulties, it is necessary to abandon the Gram-Schmidt approach and to consider obtaining the γ_{mn} from methods based on the set $\{\psi_n(\omega)\}$ derived from the Fourier transform of $\varphi_n(t)$. With the aid of Parseval's theorem and other key results from the theory of functions of a complex variable, it will be possible to achieve an explicit relation for the coefficients γ_{mn}, as well as for the transform parameters λ_{mn} in the equation

$$U_m(\omega) = \sum_{n=1}^{m} \lambda_{mn}\, \psi_n(\omega) \tag{21}$$

where

$$\int_{-\infty}^{\infty} U_r(\omega)\, U_s^*(\omega)\, d\omega = \delta_{rs} \tag{22}$$

Application of the Fourier transform, Eq. (16), to the elements $\varphi_n(t)$ and $\psi_n(\omega)$ of Eqs. (3) and (4) reveals that

$$\begin{aligned} \frac{1}{2\pi}\int_{-\infty}^{\infty} U_m(\omega)\, e^{j\omega t}\, d\omega &= \frac{1}{2\pi}\int_{-\infty}^{\infty} \sum_{n=1}^{m} \lambda_{mn}\, \psi_n(\omega)\, e^{j\omega t}\, d\omega \\ &= \sum_{n=1}^{m} \frac{\lambda_{mn}}{2\pi}\int_{-\infty}^{\infty} \psi_n(\omega)\, e^{j\omega t}\, d\omega = \sum_{n=1}^{m} \lambda_{mn}\, \varphi_n(t) \equiv \hat{X}_m(t) \end{aligned} \tag{23}$$

where m is finite. Hence, $\hat{X}_m(t)$ and $U_m(\omega)$ are Fourier transform pairs. Since $U_m(\omega)$ is assumed to be orthonormal, Parseval's theorem leads to

$$\int_{-\infty}^{\infty} U_m(\omega)\, U_n^*(\omega)\, d\omega = 2\pi \int_{-\infty}^{\infty} \hat{X}_m(t)\, \hat{X}_n^*(t)\, dt = \delta_{mn} \tag{24}$$

Consequently, if the coefficients γ_{mn} in $X_m(t)$ are chosen as

$$\gamma_{mn} = \sqrt{2\pi}\, \lambda_{mn} \tag{25}$$

then the set $\{X_m(t)\}$ will also be orthonormal, for

$$\begin{aligned}
\delta_{mn} &= 2\pi \int_{-\infty}^{\infty} \left[\sum_{n=1}^{m} \lambda_{mn}\, \varphi_n(t)\right] \left[\sum_{k=1}^{n} \lambda_{nk}^{*}\, \varphi_k^{*}(t)\right] dt \\
&= \int_{-\infty}^{\infty} \left[\sum_{n=1}^{m} \sqrt{2\pi}\, \lambda_{mn}\, \varphi_n(t)\right] \left[\sum_{k=1}^{n} \sqrt{2\pi}\, \lambda_{nk}^{*}\, \varphi_k^{*}(t)\right] dt \\
&= \int_{-\infty}^{\infty} \left[\sum_{n=1}^{m} \gamma_{mn}\, \varphi_n(t)\right] \left[\sum_{k=1}^{n} \gamma_{nk}^{*}\, \varphi_k^{*}(t)\right] dt \\
&= \int_{-\infty}^{\infty} X_m(t)\, X_n^{*}(t)\, dt
\end{aligned} \tag{26}$$

Once the orthonormal elements $\{U_m(\omega)\}$ are found, therefore, the orthonormal set $\{X_m(t)\}$ is also uniquely determined.

The condition of orthonormality and the structure of $\phi_n(\omega)$ suggest a scheme for finding the parameters λ_{mn} of $U_m(\omega)$. The procedure is based on two results of analytic function theory, and is an extension of the application of the theorems found in Refs. 1 and 5.

By observing the pole patterns of the rationalized functions $U_m(\omega)$, $m = 1,2,\ldots$, it is possible to compute the coefficients λ_{mn} which determine the m^{th} orthonormal function $U_m(\omega)$. In $U_1(\omega)$, for example,

$$U_1(\omega) = \frac{\lambda_{11}}{\sigma + j(\omega-\beta)} \tag{27}$$

a pole is located at $j\omega = -\sigma + j\beta$. In $U_2(\omega)$, where

$$
\begin{aligned}
U_2(\omega) &= \frac{\lambda_{21}}{\sigma+j(\omega-\beta)} + \frac{\lambda_{22}}{2\sigma+j(\omega-2\beta)} \\
&= \frac{\sigma(2\lambda_{21}+\lambda_{22}) + j[\omega(\lambda_{21}+\lambda_{22}) - \beta(2\lambda_{21}+\lambda_{22})]}{[\sigma+j(\omega-\beta)]\,[2\sigma+j(\omega-2\beta)]}
\end{aligned} \tag{28}
$$

poles are located at $j\omega = -\sigma+j\beta$ and $j\omega = -2\sigma+j2\beta$. In general, for

$$U_m(\omega) = \frac{\lambda_{m1}}{\sigma+j(\omega-\beta)} + \frac{\lambda_{m2}}{2\sigma+j(\omega-2\beta)} + \cdots + \frac{\lambda_{mm}}{m\sigma+j(\omega-m\beta)} \tag{29}$$

equally spaced left-half s-plane (LHP) poles are located at $j\omega = -\sigma+j\beta$, $-2\sigma+j2\beta$, ..., $-m\sigma+jm\beta$. Similarly, the conjugate function $U_m^{*}(\omega)$ has equally spaced right-half s-plane (RHP) poles at $j\omega = \sigma+j\beta$, $2\sigma+2j\beta$, ..., $m\sigma+jm\beta$. Therefore, if $U_1(\omega)$ is expressed as

$$U_1(\omega) = \frac{a_1}{\sigma+j(\omega-\beta)}\ , \quad a_1 = \text{normalizing constant} \tag{30}$$

then to orthogonalize $U_1(\omega)$ and $U_2(\omega)$, i.e., to ensure that

$$\int_{-\infty}^{\infty} U_2(\omega)\ U_1^{*}(\omega)\ d\omega = 0 \tag{31}$$

$U_2(\omega)$ must be chosen as

$$U_2(\omega) = \frac{a_2\ [\sigma-j(\omega-\beta)]}{[\sigma+j(\omega-\beta)]\ [2\sigma+j(\omega-2\beta)]}\ , \quad a_2 = \text{normalizing constant} \tag{32}$$

In this way, the zero of $U_2(\omega)$ cancels the RHP pole of $U_1^{*}(\omega)$ in the product $U_2\ U_1^{*}$, and the integrand of Eq. (31) becomes analytic in the RHP. Since $U_2\ U_1^{*}$ also satisfies Jordan's lemma,[6] the Cauchy integral theorem[7] can be applied to the integral in Eq. (31) (with the contour

of integration taken in the clockwise sense as the $j\omega$-axis and infinite RHP semicircle) to establish that

$$\int_{-\infty}^{\infty} U_2(\omega)\, U_1^*(\omega)\, d\omega = \frac{-1}{j} \oint_{-j\infty}^{j\infty} \frac{a_1^* a_2 \, ds}{[s+\sigma-j\beta]\,[s+2\sigma-j2\beta]} = 0 \tag{33}$$

Similarly, if $U_3(\omega)$ is selected as

$$U_3(\omega) = \frac{a_3[\sigma-j(\omega-\beta)]\,[2\sigma-j(\omega-2\beta)]}{[\sigma+j(\omega-\beta)]\,[2\sigma+j(\omega-2\beta)]\,[3\sigma+j(\omega-3\beta)]}\ , \quad a_3 = \text{normalizing constant} \tag{34}$$

then both $U_3\, U_2^*$ and $U_3\, U_1^*$ are analytic in the RHP, and consequently both equalities

$$\int_{-\infty}^{\infty} U_3(\omega)\, U_2^*(\omega)\, d\omega = \frac{-1}{j} \oint_{-j\infty}^{j\infty} \frac{a_2^* a_3 \, ds}{[s+2\sigma-j2\beta]\,[s+3\sigma-j3\beta]} = 0 \tag{35}$$

and

$$\int_{-\infty}^{\infty} U_3(\omega)\, U_1^*(\omega)\, d\omega = \frac{-1}{j} \oint_{-j\infty}^{j\infty} \frac{a_1^* a_3\, [-s+2\sigma+j2\beta]\, ds}{[s+\sigma-j\beta]\,[s+2\sigma-2j\beta]\,[s+3\sigma-j3\beta]} = 0 \tag{36}$$

are guaranteed by Jordan's lemma and the Cauchy integral theorem. Extending the above sequence to the m^{th} element of the orthogonal basis, it is observed that the general form of $U_m(\omega)$ must be

$$U_m(\omega) = \begin{cases} \dfrac{a_1}{\sigma+j(\omega-\beta)}\ , & m = 1 \\[2ex] \dfrac{a_m[\sigma-j(\omega-\beta)]\,[2\sigma-j(\omega-2\beta)]\cdots[(m-1)\sigma-j(\omega-m\beta+\beta)]}{[\sigma+j(\omega-\beta)]\,[2\sigma+j(\omega-2\beta)]\cdots[m\sigma+j(\omega-m\beta)]}\ , & m = 2,3,\ldots \end{cases} \tag{37}$$

where a_m is a normalizing constant.

The constants a_m are readily determined by requiring that the set $\{U_m(\omega)\}$ be orthonormal; i.e.,

$$\int_{-\infty}^{\infty} U_m(\omega)\, U_m^*(\omega)\, d\omega = 1 \tag{38}$$

Substituting Eq. (37) for $U_m(\omega)$ in the normality condition Eq. (38) gives

$$\int_{-\infty}^{\infty} d(j\omega) \left\{ \frac{|a_m|^2[\sigma-j(\omega-\beta)][2\sigma-j(\omega-2\beta)]\ldots[(m-1)\sigma-j(\omega-m\beta+\beta)]}{j\,[\sigma+j(\omega-\beta)][2\sigma+j(\omega-2\beta)]\ldots[m\sigma+j(\omega-m\beta)]} \right.$$

$$\left. \cdot \frac{[\sigma+j(\omega-\beta)][2\sigma+j(\omega-2\beta)]\ldots[(m-1)\sigma+j(\omega-m\beta+\beta)]}{[\sigma-j(\omega-\beta)][2\sigma-j(\omega-2\beta)]\ldots[m\sigma-j(\omega-m\beta)]} \right\} \tag{39}$$

$$= \int_{-\infty}^{\infty} \frac{|a_m|^2\, d(j\omega)}{j[m\sigma+j(\omega-m\beta)][m\sigma-j(\omega-m\beta)]} = 2\pi j \left\{ \frac{|a_m|^2}{j} \cdot \frac{1}{m\sigma+jm\beta-j\omega} \right\} \Bigg|_{j\omega=-m\sigma+jm\beta}$$

$$= \frac{\pi|a_m|^2}{m\sigma} = 1, \qquad m = 1,2,\ldots$$

Since the $U_m(\omega)$ are already orthogonal, the normality condition Eq. (38) can be satisfied by choosing the a_m real in Eq. (39) and by assigning to a_m the value

$$a_m = \sqrt{\frac{m\sigma}{\pi}}, \qquad m = 1,2,\ldots \tag{40}$$

In view of Eq. (37), the orthonormal basis functions can be finally written as

$$U_m(\omega) = \begin{cases} \sqrt{\dfrac{\sigma}{\pi}}\, \dfrac{1}{\sigma+j(\omega-\beta)}, & m = 1 \\[2ex] \sqrt{\dfrac{m\sigma}{\pi}}\, \dfrac{\prod\limits_{n=1}^{m-1} [n\sigma-j(\omega-n\beta)]}{\prod\limits_{n=1}^{m} [n\sigma+j(\omega-n\beta)]}, & m = 2,3,\ldots \end{cases} \tag{41}$$

The coefficients λ_{mn} in the series representation of $U_m(\omega)$, Eq. (21), can now be related to Eq. (41) by

$$\sqrt{\frac{m\sigma}{\pi}}\,\frac{[\sigma-j(\omega-\beta)][2\sigma-j(\omega-2\beta)]\cdots[(m-1)\sigma-j(\omega-m\beta+\beta)]}{[\sigma+j(\omega-\beta)][2\sigma+j(\omega-2\beta)]\cdots[m\sigma+j(\omega-m\beta)]}$$

$$= \sum_{k=1}^{m} \frac{\lambda_{mk}}{k\sigma+j(\omega-k\beta)}\,, \qquad m = 2,3,\ldots \tag{42}$$

Multiplying both sides of Eq. (42) by $n\sigma+j(\omega-n\beta)$, $1 \le n \le m$, gives

$$\sqrt{\frac{m\sigma}{\pi}}\,\frac{[n\sigma+j(\omega-n\beta)][\sigma-j(\omega-\beta)][2\sigma-j(\omega-2\beta)]\cdots[(m-1)\sigma-j(\omega-m\beta+\beta)]}{[\sigma+j(\omega-\beta)][2\sigma+j(\omega-2\beta)]\cdots[n\sigma+j(\omega-n\beta)]\cdots[m\sigma+j(\omega-m\beta)]}$$

$$= \sum_{k=1}^{m} \lambda_{mk}\left[\frac{n\sigma+j(\omega-n\beta)}{k\sigma+j(\omega-k\beta)}\right]\,, \qquad m = 2,3,\ldots;\quad 1 \le n \le m \tag{43}$$

Since Eq. (43) must be valid for all ω, it must also be an identity for $\omega = n\beta+jn\sigma$, or $j\omega = -n\sigma+jn\beta$. Accordingly, with $j\omega = -n\sigma+jn\beta$, Eq. (43) reduces to

$$\sqrt{\frac{m\sigma}{\pi}}\,\frac{\displaystyle\prod_{r=1}^{m-1}[\sigma(n+r)-j\beta(n-r)]}{\displaystyle\prod_{r=1}^{n-1}[-\sigma(n-r)+j\beta(n-r)]\prod_{r=n+1}^{m}[-\sigma(n-r)+j\beta(n-r)]} = \lambda_{mn} \tag{44}$$

so that

$$\lambda_{mn} = \begin{cases} \sqrt{\dfrac{\sigma}{\pi}} & n = m = 1 \\[2ex] \sqrt{\dfrac{m\sigma}{\pi}}\, \dfrac{\prod\limits_{r=1}^{m-1} \left[n + r\left(\dfrac{\sigma + j\beta}{\sigma - j\beta}\right)\right]}{\prod\limits_{\substack{r=1 \\ r \neq n}}^{m} [r-n]} & m = 2,3,\ldots;\ 1 \le n \le m \\[2ex] 0 & m = 1,2,\ldots;\ n > m \end{cases} \tag{45}$$ [†]

In view of Eq. (25), γ_{mn} is immediately determined for the orthonormal set $\{X_m(t)\}$ as $\gamma_{mn} = \sqrt{2\pi}\,\lambda_{mn}$.

The preceding derivation of the orthonormal basis $\{U_m(\omega)\}$ permits a simple determination of the allied real-valued functions of ω

$$V_m(\omega) = \frac{1}{\sqrt{2}}\,[U_m(\omega) + U_m^*(\omega)] = \frac{1}{\sqrt{2}} \sum_{n=1}^{m} [\lambda_{mn}\,\psi_n(\omega) + \lambda_{mn}^*\,\psi_n^*(\omega)] \tag{46}$$

Assurance of orthonormality for the set $\{V_m(\omega)\}$ follows from the relations

$$\int_{-\infty}^{\infty} V_m(\omega)\, V_n^*(\omega)\, d\omega \equiv \frac{1}{2} \int_{-\infty}^{\infty} [U_m + U_m^*]\,[U_n^* + U_n]\, d\omega \tag{47}$$

or

$$\int_{-\infty}^{\infty} V_m(\omega)\, V_n^*\, d\omega = \frac{1}{2}\int_{-\infty}^{\infty} U_m U_n^*\, d\omega + \frac{1}{2}\int_{-\infty}^{\infty} U_m^* U_n\, d\omega + \frac{1}{2}\int_{-\infty}^{\infty} U_m U_n\, d\omega + \frac{1}{2}\int_{-\infty}^{\infty} U_m^* U_n^*\, d\omega \tag{48}$$

[†] For the functions $\hat{\psi}_n(\omega)$ defined in Eq. (2), the result for λ_{mn} with $\beta = 0$ agrees with the quantity $\sqrt{\frac{m\sigma}{\pi}}\,\alpha_{mn}$ derived in Ref. 1.

Since the U_m are orthonormal (Eqs. (33-38)) the first two integrals on the right side of Eq. (48) yield the Kronecker delta

$$\int_{-\infty}^{\infty} V_m V_n^* \, d\omega = \delta_{mn} + \int_{-\infty}^{\infty} U_m U_n \, d\omega + \frac{1}{2}\int_{-\infty}^{\infty} U_m^* U_n^* \, d\omega \tag{49}$$

It is evident from Eq. (37) for $U_m(\omega)$ that the integrand $U_m U_n$ in the first integral of Eq. (49) contains only LHP poles. Since Jordan's lemma[†] is satisfied for $U_m U_n$, the first integral of Eq. (49) may be evaluated over the infinite RHP semicircle with diameter on the $j\omega$-axis. Since $U_m U_n$ is also holomorphic in this RHP region, Cauchy's integral theorem[††] guarantees that the integral of $U_m U_n$ vanishes. Similarly, with $U_m^* U_n^*$ holomorphic in the entire LHP, it can be argued that the second integral of Eq. (49) is also zero. Consequently, Eq. (49) becomes

$$\int_{-\infty}^{\infty} V_m(\omega) \, V_n^*(\omega) \, d\omega = \delta_{mn} \tag{50}$$

The rational function expression of V_m may be written with the aid of Eqs. (46), (21), and (4) as

$$V_m(\omega) = \frac{1}{\sqrt{2}} \sum_{n=1}^{m} \frac{(\lambda_{mn}+\lambda_{mn}^*)\, n\sigma + j(-\lambda_{mn}+\lambda_{mn}^*)(\omega-n\beta)}{(n\sigma)^2 + (\omega-n\beta)^2} \tag{51}$$

or

$$V_m(\omega) = \sqrt{2} \sum_{n=1}^{m} a_{mn} \frac{n\sigma}{(n\sigma)^2 + (\omega-n\beta)^2} + \sqrt{2} \sum_{n=1}^{m} b_{mn} \frac{\omega-n\beta}{(n\sigma)^2 + (\omega-n\beta)^2} \tag{52}$$ [†††]

[†] See Ref. 6.

[††] See Ref. 7.

[†††] For $\beta = 0$, λ_{mn} is real and $b_{mn} = 0$. When $\sqrt{2}\, a_{mn}$ is also an integer, m,n = 1,2,..., $V_m(\omega)$ takes the form of the envelope-delay components examined in Ref. 1.

where a_{mn} and b_{mn} are related to λ_{mn}, Eq. (45), as

$$\lambda_{mn} \equiv a_{mn} + jb_{mn} \tag{53}$$

Thus, Eq. (52) can be rationalized to give

$$V_m(\omega) \begin{cases} \sqrt{\frac{2\sigma}{\pi}} \dfrac{\sigma}{\sigma^2+(\omega-\beta)^2}, & m = 1 \\ \sqrt{\frac{2m\sigma}{\pi}} \operatorname{Re} \left\{ \dfrac{[m(\sigma+j\beta) - j\omega] \prod\limits_{n=1}^{m-1} [n(\sigma+j\beta) - j\omega]^2}{\prod\limits_{n=1}^{m} [(n\sigma)^2 + (\omega-n\beta)^2]} \right\}, & m = 2,3,\ldots \end{cases} \tag{54}$$

Similarly, in the transformed space, it can be shown that the functions

$$Y_m(t) = \frac{1}{\sqrt{2}} [X_m(t) + X_m^*(-t)] = \sqrt{\pi} \left[\sum_{n=1}^{m} \lambda_{mn} \varphi_n(t) + \sum_{n=1}^{m} \lambda_{mn}^* \varphi_n^*(-t) \right] \tag{55}$$

comprise an orthonormal basis. Equation (16) indicates that $\varphi_n(t)$ and $\psi_n(\omega)$, as defined in Eqs. (3) and (4), are Fourier transform pairs. Thus,

$$\varphi_n(t) = \frac{1}{2\pi} \int_{-\infty}^{\infty} \psi_n(\omega) e^{j\omega t} d\omega \tag{56}$$

and

$$\varphi_n^*(-t) = \frac{1}{2\pi} \int_{-\infty}^{\infty} \psi_n^*(\omega) e^{j\omega t} d\omega \tag{57}$$

Substituting these relations into the expression for $V_m(\omega)$,

$$V_m(\omega) = \frac{1}{\sqrt{2}} [U_m(\omega) + U_m^*(\omega)] = \frac{1}{\sqrt{2}} \left[\sum_{n=1}^{m} \lambda_{mn} \psi_n(\omega) + \sum_{n=1}^{m} \lambda_{mn}^* \psi_{mn}^*(\omega) \right] \tag{46}$$

it follows that the Fourier transform of $V_m(\omega)$ is

$$\frac{1}{2\pi} \int_{-\infty}^{\infty} V_m(\omega) e^{j\omega t} d\omega = \frac{1}{\sqrt{2}} \left[\sum_{n=1}^{m} \lambda_{mn} \varphi_n(t) + \sum_{n=1}^{m} \lambda_{mn}^* \varphi_n^*(-t) \right] = \sqrt{\frac{1}{2\pi}} Y_m(t) \tag{58}$$

Consequently, Parseval's theorem and the orthonormality relation, Eq. (50), for $V_m(\omega)$ allow Eq. (50) to be written as

$$\int_0^{\infty} \frac{Y_m(t)}{\sqrt{2\pi}} \frac{Y_n^*(t)}{\sqrt{2\pi}} dt = \frac{1}{2\pi} \int_{-\infty}^{\infty} V_m(\omega) V_n^*(\omega) d\omega = \frac{1}{2\pi} \delta_{mn} \tag{59}$$

so that

$$\int_0^{\infty} Y_m(t) Y_n^*(t) dt = \delta_{mn} \tag{60}$$

From Eqs. (55), (53), (45), and (3), the orthonormal basis functions $Y_m(t)$ can be finally expressed as the complex-valued functions of t

$$Y_m(t) = \begin{cases} \sqrt{\pi} \sum_{n=1}^{m} \lambda_{mn} e^{-n(\sigma - j\beta)t} & t > 0 \\ & m = 1,2,\ldots \\ \sqrt{\pi} \sum_{n=1}^{m} \lambda_{mn}^* e^{+n(\sigma + j\beta)t} & t < 0 \end{cases} \tag{61}$$

or, expressing λ_{mn} in polar form,

$$Y_m(t) = \sqrt{\pi} \sum_{n=1}^{m} |\lambda_{mn}| \, e^{-n\sigma|t|} \left\{ \cos\left[n\beta t + \frac{t}{|t|} \arg(\lambda_{mn}) \right] + \frac{t}{|t|} \, j \sin\left[n\beta|t| + \arg(\lambda_{mn}) \right] \right\} \quad m = 1,2,\ldots \quad \forall\, t \tag{62}$$

The key relations established for the orthonormal bases $X_m(t)$ and $U_m(\omega)$ and the equations determining the orthonormal basis coefficients λ_{mn} are summarized in Table 1.

Table 1[a]

EQUIVALENT REPRESENTATIONS OF $X_m(t)$, $U_m(\omega)$, and λ_{mn}

$X_m(t)$	$U_m(\omega)$	λ_{mn}
	Range	
$0 < t < \infty \quad \left[X_m(t) = 0, \quad t \leq 0\right]$	$-\infty < \omega < \infty$	$m = 2,3,\ldots$ $n = 1,2,\ldots,m$ $\quad \lambda_{11} = 1$
$\sigma > 0$, $\mathrm{Im}(\sigma) = 0$, $\mathrm{Im}(\beta) = 0$	$\sigma > 0$, $\mathrm{Im}(\sigma) = 0$, $\mathrm{Im}(\beta) = 0$	$\lambda_{mn} = 0 \qquad n > m$
	Representation	
$\sum_{n=1}^{m} \gamma_{mn}\, \varphi_n(t)$	$\sum_{n=1}^{m} \lambda_{mn}\, \psi_n(\omega)$	$\sqrt{\frac{m\sigma}{\pi}}\, \frac{\prod_{r=1}^{m-1}\left[n+r\left(\frac{\sigma+j\beta}{\sigma-j\beta}\right)\right]}{\prod_{r=1,\, r\neq n}^{m} [r-n]}$
$\sum_{n=1}^{m} \gamma_{mn}\, e^{-n(\sigma-j\beta)t}$	$\sum_{n=1}^{m} \lambda_{mn}\, \frac{1}{n\sigma+j(\omega-n\beta)}$	$\sqrt{\frac{m\sigma}{\pi}}\, \frac{\prod_{r=1}^{m-1}[n(\sigma-j\beta) + r(\sigma+j\beta)]}{(-1)^{n+1}(n-1)!(m-n)!(\sigma-j\beta)^{m-1}}$
$\sum_{n=1}^{m} \sqrt{2\pi}\, \lambda_{mn}\, e^{-n(\sigma-j\beta)t}$	$\sum_{n=1}^{m} \sqrt{\frac{m\sigma}{\pi}}\, \alpha_{mn}\, \frac{1}{n\sigma+j(\omega-n\beta)}$	$\sqrt{\frac{m\sigma}{\pi}}\, \frac{(-1)^{m+n+1}}{n!} \sum_{k=0}^{m} (-1)^k \binom{m}{n} S_m^{(k)} n^k \left[\frac{\sigma+j\beta}{\sigma-j\beta}\right]^{m-k}$
$\sqrt{2\pi}\, \mathfrak{F}^{-1}\{U_m(\omega)\}$	$\sqrt{\frac{1}{2\pi}}\, \mathfrak{F}\{X_m(t)\}$	$\sqrt{\frac{m}{m-1}}\, \frac{n(\sigma-j\beta) + (m-1)(\sigma+j\beta)}{(m-n)(\sigma-j\beta)}\, \lambda_{m-1,n} \qquad n < m$

[a]The last two relations in the λ_{mn} column are derived in suceeding sections. $\mathfrak{F}$ denotes the Fourier transform operator, and $\mathfrak{F}^{-1}$ denotes the inverse Fourier transform operator (see Eq. (23)). The quantity α_{mn} is defined by Eq. (68).

V. SPECIAL PROPERTIES OF THE BASIS COEFFICIENTS λ_{mn} AND THE ORTHONORMAL ELEMENTS $X_m(t)$, $U_m(\omega)$

The preceding analysis culminates in Eq. (45) for the basis coefficients λ_{mn} which generate the orthonormal elements $U_m(\omega)$, $V_m(\omega)$, $X_m(t)$, and $Y_m(t)$. In order to detect errors in the computation of these λ_{mn}, it is desirable to provide a check-sum relation for the λ_{mn} analogous to the expression for the coefficients α_{mn} of Ref. 1.

It is shown in Ref. 1 that for $\beta = 0$ an identity

$$\sum_{n=1}^{m} \hat{\alpha}_{mn} = (-1)^{m+1} \qquad m = 1,2,... \tag{63}$$[†]

exists in which the α_{mn} are given by

$$\hat{\alpha}_{mn} = \begin{cases} (-1)^{n-1} \binom{m-1}{n-1} \binom{m+n-1}{m-1} & 1 \le n \le m = 1,2,... \\ \\ 0 & n > m \end{cases} \tag{64}$$[††]

The $\hat{\alpha}_{mn}$ are related to the orthonormal elements $R_m(t)$ or $W_m(\omega)$ by

$$R_m(t) = \sum_{n=1}^{m} \sqrt{2m\sigma}\, \hat{\alpha}_{mn}\, e^{-n\sigma t} \qquad 0 < t < \infty \quad m = 1,2,... \tag{65}$$[†††]

and

$$W_m(\omega) = \sum_{n=1}^{m} \sqrt{\frac{m\sigma}{\pi}}\, \hat{\alpha}_{mn} \frac{1}{n\sigma + j\omega} \qquad |\omega| < \infty \qquad m = 1,2,... \tag{66}$$

[†] See Ref. 1, p. 22.

[††] See Ref. 1, p. 21.

[†††] See Ref. 1, Table 1.

For the λ_{mn} derived in this Memorandum, it is similarly demonstrable that

$$\sqrt{\frac{\pi}{m\sigma}}\sum_{n=1}^{m} \lambda_{mn} = \sum_{n=1}^{m} \alpha_{mn} = (-1)^{m+1} \qquad m = 1,2,\ldots \tag{67}$$

where α_{mn} is defined as

$$\alpha_{mn} \equiv \sqrt{\frac{\pi}{m\sigma}}\,\lambda_{mn} = \begin{cases} 1 & n = m = 1 \\ \dfrac{\prod_{r=1}^{m-1}\left[n + r\left(\frac{\sigma+j\beta}{\sigma-j\beta}\right)\right]}{\prod_{r=1}^{m}{}' [r-n]} & m = 2,3,\ldots;\quad 1 \leq n \leq m \\ 0 & m = 1,2,\ldots;\quad n > m \end{cases} \tag{68}$$[†]

For the special case in which $\beta = 0$, it is clear that $\alpha_{mn} = \hat{\alpha}_{mn}$, and the validity of Eq. (67) follows directly from the identity Eq. (63).[††] In general, for arbitrary real β, it must be established that

$$\sum_{n=1}^{m} \alpha_{mn} \equiv \sum_{n=1}^{m} \frac{\prod_{r=1}^{m-1}\left[n + r\left(\frac{\sigma+j\beta}{\sigma-j\beta}\right)\right]}{\prod_{r=1}^{m}{}' [r - n]} = (-1)^{m+1} \tag{69}$$

or that

$$\sum_{n=1}^{m} \frac{\prod_{r=1}^{m-1} (n + rz)}{\prod_{r=1}^{m}{}' (r - n)} = (-1)^{m+1} \tag{70}$$

[†] The prime used in the product notation $\prod_{r=1}^{m}{}' (r-n)$ signifies that r ranges only over the integers $1,2,\ldots,n-1,n+1,\ldots,m$ so that the result is never zero.

[††] See Ref. 1, p. 20, Eq. (64).

where, for convenience, the complex variable z is defined as

$$z = \frac{\alpha + j\beta}{\alpha - j\beta} \tag{71}$$

From the definition of the Stirling numbers of the first kind,[8] it is possible to write the factorial function relation

$$(x-1)(x-2)\cdots[x-(m-1)] = \sum_{n=0}^{m} S_m^{(n)}\, x^{n-1} \tag{72}$$

or

$$(x+1)(x+2)\cdots[x+(m-1)] = \sum_{n=0}^{m} S_m^{(n)}\, x^{n-1}\, (-1)^{n+m} \tag{73}$$

Hence, the numerators of Eq. (70) can be written as

$$\prod_{r=1}^{m-1} (n+rz) = \prod_{r=1}^{m-1} z\left(\frac{n}{z}+r\right) = z^{m-1} \sum_{k=0}^{m} S_m^{(k)} \left(\frac{n}{z}\right)^{k-1} (-1)^{k+m} \tag{74}$$

and the left side of Eq. (70) becomes

$$\sum_{n=1}^{m} \frac{\prod_{r=1}^{m-1}(n+rz)}{\prod_{r=1}^{m}{}'\,(r-n)} = \sum_{n=1}^{m} \frac{z^{m-1}}{\prod_{r=1}^{m}{}'\,(r-n)} \sum_{k=0}^{m} S_m^{(k)} \left(\frac{n}{z}\right)^{k-1} (-1)^{k+m} = \sum_{n=1}^{m} a_{mn} \tag{75}$$

In addition, the denominators of Eq. (70) can be expressed as

$$\prod_{r=1}^{m}{}'\,(r-n) = (-1)^{m-1} \prod_{r=1}^{m}{}'\,(n-r) = (-1)^{2m-n-1}\,\frac{n!(m-n)!}{n}, \quad 1 \le n \le m \tag{76}$$

or

$$\prod_{r=1}^{m}{}'\,(r-n) = (-1)^{n+1}\,\frac{n!(m-n)!}{n} = (-1)^{n+1}\,\frac{m!}{n\binom{m}{n}} \tag{77}$$

so that Eqs. (69), (70), and (75) can be formulated as

$$\sum_{n=1}^{m} \alpha_{mn} = \frac{(-1)^{m+1}}{m!} \sum_{n=1}^{m} \sum_{k=0}^{m} (-1)^{k+n} \binom{m}{n} S_m^{(k)} n^k z^{m-k} \tag{78}$$

From this last relation, it is clear that Eq. (69) can be established as an identity if it can be verified that

$$\frac{(-1)^{m+1}}{m!} \sum_{k=0}^{m} \sum_{n=1}^{m} (-1)^{k+n} \binom{m}{n} S_m^{(k)} n^k z^{m-k} = (-1)^{m+1} \tag{79}$$

or, equivalently, that

$$\sum_{k=0}^{m} \frac{(-1)^k S_m^{(k)}}{m!} \left\{ \sum_{n=1}^{m} (-1)^n \binom{m}{n} n^k \right\} z^{m-k} - 1 = 0 \qquad m = 1,2,\ldots \tag{80}$$

Since Eq. (80) is a polynomial of degree m in z, it follows from the fundamental theorem of algebra[9] that Eq. (80) has only m values of z which make the left side equal to zero. In order for Eq. (80) to be generally valid, therefore, each coefficient of z^r, $r = 1,2,\ldots,m$, must be zero and the coefficient of z^0 must be unity (since $S_m^{(k)}/m!$ is also nonzero for $k = 1,2,\ldots,m$; $m = 1,2,\ldots$). Consequently, the proof of Eq. (69) rests on demonstrating that

$$\frac{(-1)^{m-k} S_m^{(m-k)}}{m!} \sum_{n=1}^{m} (-1)^n \binom{m}{n} n^{m-k} = \begin{cases} 0 & k = 1,2,\ldots,m \\ 1 & k = 0 \end{cases} \tag{81}$$

or, since $S_m^{(0)} = 0$, $S_m^{(m)} = 1$, and $S_m^{(k)} \neq 0$, $k = 1,2,\ldots,m$, $m = 1,2,\ldots$, that

$$\sum_{n=1}^{m} (-1)^n \binom{m}{n} n^{m-k} = \sum_{n=0}^{m} (-1)^n \binom{m}{n} n^{m-k} = \begin{cases} 0 & k = 1,2,\ldots,m-1 \\ (-1)^m\, m! & k = 0 \end{cases} \tag{82}$$

In order to verify Eq. (82), the following identity is utilized:

$$\Delta^m f(x) = \sum_{n=0}^{m} (-1)^{m-n} \binom{m}{n} f(x+n) \tag{83}$$ [†]

where Δ denotes the forward difference operator. With the choice

$$f(x) = x^{m-k} \qquad k = 0,1,\ldots,m; \quad m = 1,2,\ldots \tag{84}$$

Eq. (83) becomes

$$\Delta^m\, x^{m-k} = \sum_{n=0}^{m} (-1)^{m-n} \binom{m}{n} (x+n)^{m-k} \tag{85}$$

It is possible, now, to conclude the demonstration by applying to Eq. (85) the fundamental theorem of difference calculus:[10]

The n^{th} difference of a polynomial of degree n

$$y(x) = \sum_{j=0}^{n} a_j\, x^j \qquad a_n \neq 0$$

is a constant, $a_n\, n!\, h$, and the $(n+1)^{th}$ difference is equal to zero. The first forward difference is defined as $\Delta y(x) \equiv y(x+h) - y(x)$; the second forward difference, as $\Delta^2 y(x) = \Delta y(x+h) - \Delta y(x)$; etc., and h is a constant.

Accordingly, for $k = 1,2,\ldots,m-1$

$$\Delta^m\, x^{m-k} = 0 \qquad k = 1,2,\ldots,m-1 \tag{86}$$

[†] See Ref. 8, p. 823.

so that Eq. (85) becomes

$$\sum_{n=0}^{m} (-1)^{m-n} \binom{m}{n} (x+n)^{m-k} = 0 \qquad k = 1,2,\ldots,m-1 \tag{87}$$

For the particular choice x = 0, Eq. (87) becomes

$$\sum_{n=0}^{m} (-1)^{n} \binom{m}{n} n^{m-k} = 0 \qquad k = 1,2,\ldots,m-1 \tag{88}$$

which verifies Eq. (82) or Eq. (81) for all $k = 1,2,\ldots,m$ except $k = 0$. With $k = 0$, $h = 1$, Eq. (85) may be simplified through the finite difference theorem to

$$\Delta^{m} x^{m} = m! = \sum_{n=0}^{m} (-1)^{m-n} \binom{m}{n} (x+n)^{m} \tag{89}$$

or, with x = 0

$$\sum_{n=0}^{m} (-1)^{n} \binom{m}{n} n^{m} = (-1)^{m}\, m! \tag{90}$$

Since this last relation is recognized as Eq. (82) with $k = 0$, Eq. (82) and, therefore, Eq. (69) are established as identities for $m = 1,2,\ldots$; $k = 0,1,\ldots,m$. From Eqs. (68) and (69), it is clear that the coefficients λ_{mn} satisfy the check-sum expression

$$\sum_{n=1}^{m} \lambda_{mn} = (-1)^{m+1}\sqrt{\frac{m\sigma}{\pi}} \qquad m = 1,2,\ldots \tag{91}$$

Two additional derivations simplify the calculation of the initial values of the basis elements $X_m(t)$ and $U_m(\omega)$, as well as the evaluation of the integrals of these functions. In order to get these relations, it is necessary to verify the identity

$$m\left[\frac{\sigma - j\beta}{\sigma + j\beta}\right]^{m-1} \sum_{n=1}^{m} \frac{\alpha_{mn}}{n} = 1 \tag{92}$$

or, using the definition of z in Eq. (71), to show that

$$m\, z^{(1-m)} \sum_{n=1}^{m} \frac{\alpha_{mn}}{n} = 1 \tag{93}$$

By referring to Eq. (78), it is seen that Eq. (93) can be written as

$$\frac{m(-1)^{m+1}}{z^{m-1}\, m!} \sum_{n=1}^{m} \sum_{k=0}^{m} (-1)^{k+n} \binom{m}{n} S_m^{(k)}\, n^{k-1}\, z^{m-k} = 1 \tag{94}$$

or, reordering the summation,

$$\sum_{k=0}^{m} S_m^{(k)} \frac{(-1)^{m+1+k}}{(m-1)!} \sum_{n=1}^{m} (-1)^n \binom{m}{n} n^{k-1}\, z^{m-k} = z^{m-1} \tag{95}$$

Since $S_m^{(0)} = 0$, it is necessary to show that

$$\sum_{k=1}^{m} S_m^{(k)} \frac{(-1)^{m+1+k}}{(m-1)!} \left\{ \sum_{n=1}^{m} (-1)^n \binom{m}{n} n^{k-1} \right\} z^{m-k} - z^{m-1} = 0 \tag{96}$$

or, applying again the fundamental theorem of algebra, that

$$S_m^{(1)} \frac{(-1)^m}{(m-1)!} \sum_{n=1}^{m} (-1)^n \binom{m}{n} = 1 \tag{97}$$

and, since $S_m^{(k)} \neq 0$, $k = 1,2,\ldots,m$, that

$$\sum_{n=1}^{m} (-1)^n \binom{m}{n} n^{k-1} = 0 \qquad k = 2,3,\ldots,m \tag{98}$$

Equation (98) follows immediately from the finite difference relation given in Eq. (83):

$$\Delta^m f(x) = \sum_{n=0}^{m} (-1)^{m-n} \binom{m}{n} f(x+n)$$

If f(x) is chosen as

$$f(x) = x^{k-1} \tag{99}$$

then Eq. (83) states that

$$\Delta^m x^{k-1} = \sum_{n=0}^{m} (-1)^{m-n} \binom{m}{n} (x+n)^{k-1} \tag{100}$$

The previously cited theorem of difference calculus provides the relation

$$\Delta^m x^{k-1} = 0 \qquad k = 2,3,\ldots,m \tag{101}$$

Consequently, Eq. (100) becomes

$$\sum_{n=0}^{m} (-1)^{m-n} \binom{m}{n} (x+n)^{k-1} = 0 \qquad k = 2,3,\ldots,m \tag{102}$$

With the choice x = 0, this simplifies to

$$\sum_{n=0}^{m} (-1)^{n} \binom{m}{n} n^{k-1} = 0 \qquad k = 2,3,\ldots,m \tag{103}$$

or, since the first term of the summand is zero

$$\sum_{n=1}^{m} (-1)^{n} \binom{m}{n} n^{k-1} = 0 \qquad k = 2,3,\ldots,m \tag{104}$$

This last equation is recognized as Eq. (98).

It remains to demonstrate the validity of Eq. (97). Since $S_m^{(1)}$ is given by

$$S_m^{(1)} = (-1)^{m-1}\,(m-1)! \qquad m = 1,2,\ldots \tag{105}$$ [†]

Eq. (97) becomes

$$-\sum_{n=1}^{m} (-1)^n \binom{m}{n} = 1 \tag{106}$$

or, adding and subtracting the term for n = 0,

$$\sum_{n=0}^{m} (-1)^n \binom{m}{n} = 0 \qquad m = 1,2,\ldots \tag{107}$$

But Eq. (107) follows directly from the binomial expansion[(11)]

$$(1 + x)^m = \sum_{n=0}^{m} \binom{m}{n} x^n \qquad m = 1,2,\ldots \tag{108}$$

Thus, with x = -1, Eq. (108) yields

$$0 = \sum_{n=0}^{m} \binom{m}{n} (-1)^n \qquad m = 1,2,\ldots \tag{109}$$

Since this is precisely Eq. (107), Eq. (97) is verified, and thereby, the identity Eq. (92). Finally, utilizing Eqs. (92) and (68), it follows that

$$\sqrt{\frac{m\pi}{\sigma}} \left[\frac{\sigma - j\beta}{\sigma + j\beta}\right]^{m-1} \sum_{n=1}^{m} \frac{\lambda_{mn}}{n} = 1 \tag{110}$$

The identities given by Eqs. (91) and (110) can be used to evaluate $X_m(0)$, $U_m(0)$, $\int_0^\infty X_m(t)\,dt$, and $\int_{-\infty}^{\infty} U_m(\omega)\,d\omega$. From Eqs. (18) and (25), $X_m(t)$ becomes

---

[†] See Ref. 8, p. 824.

$$X_m(t) = \sum_{n=1}^{m} \sqrt{2\pi}\, \lambda_{mn}\, \varphi_n(t) \qquad m = 1,2,\ldots \tag{111}$$

This can also be expressed, by Eq. (3) for $\varphi_n(t)$, as

$$X_m(t) = \sqrt{2\pi} \sum_{n=1}^{m} \lambda_{mn}\, e^{-n(\sigma-j\beta)t} \qquad m = 1,2,\ldots;\quad t > 0 \tag{112}$$

Hence, $X_m(0)$ becomes

$$X_m(0) = \sqrt{2\pi} \sum_{n=1}^{m} \lambda_{mn} = (-1)^{m+1} \sqrt{2m\sigma} \qquad m = 1,2,\ldots \tag{113}$$

Similarly, using Eqs. (4) and (21), $U_m(\omega)$ can be written as

$$U_m(\omega) = \sum_{n=1}^{m} \lambda_{mn} \left[\frac{1}{n\sigma + j(\omega-n\beta)}\right] \qquad m = 1,2,\ldots;\quad |\omega| < \infty \tag{114}$$

so that

$$U_m(0) = \frac{1}{\sigma - j\beta} \sum_{n=1}^{m} \frac{\lambda_{mn}}{n} \qquad m = 1,2,\ldots \tag{115}$$

or, in view of Eq. (110)

$$U_m(0) = \sqrt{\frac{\sigma}{m\pi}}\, \frac{(\sigma + j\beta)^{m-1}}{(\sigma - j\beta)^m} \qquad m = 1,2,\ldots \tag{116}$$

The areas under $U_m(\omega)$ and $X_m(t)$ can be ascertained in a similar fashion. From Eq. (23)

$$\frac{1}{2\pi} \int_{-\infty}^{\infty} U_m(\omega)\, e^{j\omega t}\, d\omega = \hat{X}_m(t) = \sum_{n=1}^{m} \lambda_{mn}\, \varphi_n(t)$$

If $\varphi_n(t)$ is substituted in Eq. (23), it follows that

$$\frac{1}{2\pi}\int_{-\infty}^{\infty} U_m(\omega)\ d\omega = \sum_{n=1}^{m} \lambda_{mn}\ \varphi_n(0) = \sum_{n=1}^{m} \lambda_{mn} \tag{117}$$

Consequently, replacing the sum of the λ_{mn} with the right side of Eq. (91), the area under $U_m(\omega)$ becomes

$$\int_{-\infty}^{\infty} U_m(\omega)\ d\omega = (-1)^{m+1}\sqrt{4\pi m\sigma} \qquad m = 1,2,\ldots \tag{118}$$

Since Eq. (23) relates $U_m(\omega)$ and $\hat{X}_m(t)$ as Fourier transform pairs, it is evident that

$$U_m(\omega) = \int_0^{\infty} \hat{X}_m(t)\ e^{-j\omega t}\ dt = \frac{1}{\sqrt{2\pi}}\int_0^{\infty} X_m(t)\ e^{-j\omega t}\ dt \tag{119}$$

Using Eqs. (21) and (4) for $U_m(\omega)$ and $\psi_n(\omega)$, Eq. (119) becomes

$$\frac{1}{\sqrt{2\pi}}\int_0^{\infty} X_m(t)\ e^{-j\omega t}\ dt = \sum_{n=1}^{m} \lambda_{mn}\left[\frac{1}{n\sigma + j(\omega-n\beta)}\right] \tag{120}$$

Setting $\omega = 0$ in this last equation yields

$$\int_0^{\infty} X_m(t)\ dt = \frac{\sqrt{2\pi}}{\sigma-j\beta}\sum_{n=1}^{m}\frac{\lambda_{mn}}{n} \tag{121}$$

and, utilizing the identity Eq. (110)

$$\int_0^{\infty} X_m(t)\ dt = \sqrt{\frac{2\sigma}{m}}\ \frac{(\sigma+j\beta)^{m-1}}{(\sigma-j\beta)^{m}} \qquad m = 1,2,\ldots \tag{122}$$

The important identities derived in this section are summarized in Table 2.

Table 2

SPECIAL PROPERTIES OF $X_m(t)$, $U_m(\omega)$ and λ_{mn}

$X_m(t)$	$U_m(\omega)$	λ_{mn}
Range		
$m = 1,2,\ldots$	$m = 1,2,\ldots$	$m = 1,2,\ldots$ $n = 1,2,\ldots,m$
Property		
$\int_0^\infty X_m(t)\,dt = \sqrt{\frac{2\sigma}{m}}\frac{(\sigma+j\beta)^{m-1}}{(\sigma-j\beta)^m}$	$\int_{-\infty}^\infty U_m(\omega)\,d\omega = (-1)^{m+1}\sqrt{4\pi m\sigma}$	$\sum_{n=1}^m \lambda_{mn} = (-1)^{m+1}\sqrt{\frac{m\sigma}{\pi}}$
$X_m(0) = (-1)^{m+1}\sqrt{2m\sigma}$	$U_m(0) = \sqrt{\frac{\sigma}{m\pi}}\frac{(\sigma+j\beta)^{m-1}}{(\sigma-j\beta)^m}$	$\sum_{n=1}^m \frac{\lambda_{mn}}{n} = \left[\frac{\sigma+j\beta}{\sigma-j\beta}\right]^{m-1}\sqrt{\frac{\sigma}{m\pi}}$
$\int_0^\infty X_m(t)\,X_n^*(t)\,dt = \delta_{mn}$	$\int_{-\infty}^\infty U_m(\omega)\,U_n^*(\omega)\,d\omega = \delta_{mn}$	$\lambda_{mn} = \sqrt{\frac{m\sigma}{\pi}}\frac{(-1)^{m+n+1}}{n!(m-n)!}\sum_{k=0}^m (-1)^k S_m^{(k)} n^k \left[\frac{\sigma+j\beta}{\sigma-j\beta}\right]^{m-k}$
$\sqrt{2\pi}\,\mathfrak{F}^{-1}\{U_m(\omega)\}$	$\sqrt{\frac{1}{2\pi}}\,\mathfrak{F}\{X_m(t)\}$	$\lambda_{mn} = \sqrt{\frac{m\sigma}{\pi}}\frac{(-1)^{m+n+1}}{(\sigma-j\beta)^{m-1}}\sum_{k=1}^m\sum_{r=1}^{m-k+1}\sum_{p=1}^k Q_{mn}(k,r,p)\sigma^{(m-r-p+1)}\beta^{(r+p-2)}$†

†See Appendix E. $Q_{mn}(k,r,p) \equiv \left[(-1)^{p+k+1}S_m^{(k)}n^k(m-k)!(k-1)!j^{(r+p)}\right] \div \left[(r-1)!(p-1)!(k-p)!(m-k-r+1)!\right]$.

VI. REPRESENTATIONS WITH $X_m(t)$ AND $U_m(\omega)$

The preceding sections have focused on generating the complete orthonormal exponential sums $X_m(t)$ and $Y_m(t)$ and the rational functions $U_m(\omega)$ and $V_m(\omega)$. Because the corresponding elements of these sets are related as Fourier transform pairs, it is a simple matter to determine simultaneously the least-mean-squared-error representations of prescribed functions $g(t) \subset L^2(0,\infty)$ by $X_m(t)$ or $Y_m(t)$, and of functions $h(\omega) \subset L^2(-\infty,\infty)$ by $U_m(\omega)$ or $V_m(\omega)$. The solution of the optimum expansion coefficients follows the usual treatment found in the literature on generalized Fourier analysis. Certain noteworthy simplifications evolve, however, because of the special nature of the orthonormal functions developed in this Memorandum.

When a specified function $g(x) \subset L^2(0,\infty)$ is to be approximated in a least-integrated-squared-error sense by orthonormal elements $\{\theta_m(x)\}$ complete in $L^2(0,\infty)$, it is necessary to determine the coefficients a_m in the M^{th} partial sum

$$g(x) \approx \sum_{m=1}^{M} a_m \theta_m(x) \equiv \hat{g}(x) \qquad x > 0 \tag{123}$$

so that the L^2-norm

$$\|g(x) - \hat{g}(x)\|^2 \equiv \int_0^\infty |g(x) - \hat{g}(x)|^2 \, dx \tag{124}$$

is minimized. The necessary condition for achieving an extremum of Eq. (124) is that[12]

$$\frac{\partial \|g(x) - \hat{g}(x)\|^2}{\partial a_r} = 0 \qquad r = 1,2,\ldots,M \qquad (125)$$

or, substituting Eq. (123) for g(x) in Eq. (125), that

$$\frac{\partial}{\partial a_r}\int_0^\infty \left[g(x)\ g^*(x) - g(x)\sum_{m=1}^{M} a_m^*\ \theta_m^*(x) - g^*(x)\sum_{m=1}^{M} a_m\ \theta_m(x) + \sum_{m=1}^{M}\sum_{n=1}^{M} a_m\ a_n^*\ \theta_m(x)\ \theta_n^*(x)\right] dx = 0 \qquad r = 1,2,\ldots,M \qquad (126)$$

Upon differentiating Eq. (126) and utilizing the orthonormality of the $\theta_m(x)$, Eq. (126) reduces to

$$-\int_0^\infty g(x)\ \theta_r^*(x)dx - \int_0^\infty g^*(x)\ \theta_r(x)dx + a_r^* + a_r = 0 \qquad (127)$$

This last equation in a_r is obviously satisfied by

$$a_r = \int_0^\infty g(x)\ \theta_r^*(x)\ dx \qquad r = 1,2,\ldots,M \qquad (128)$$

The fact that this value of a_r leads to the desired minimization of the norm $\|g(x) - \hat{g}(x)\|^2$ can be seen from the sufficiency condition[13]

$$\frac{\partial^2 \|g(x) - \hat{g}(x)\|^2}{\partial a_r^2} = 2 > 0 \qquad r = 1,2,\ldots,M \qquad (129)$$

For the particular set of functions $R_m(t)$ given by Eq. (18), it is clear from Eq. (128) that a least-squared-error representation $\hat{g}(t)$ of a function $g(t) \subset L^2(0,\infty)$ of the form

$$g(t) \approx \sum_{m=1}^{M} a_m\ X_m(t) \equiv \hat{g}(t) \qquad 0 < t < \infty \qquad (130)$$

requires that the a_m be selected according to

$$a_m = \int_0^\infty g(t)\, X_m^*(t)\, dt = \sum_{n=1}^{m} \sqrt{2\pi}\, \lambda_{mn}^* \int_0^\infty g(t)\, e^{-n(\sigma-j\beta)t}\, dt \qquad m = 1,2,\ldots,M \tag{131}$$

The integrals determining these a_m are identifiable as the Laplace transform[14] of $g(t)$ evaluated at the complex frequencies $s_n = n(\sigma-j\beta)$, $n = 1,2,\ldots,M$, i.e.,

$$\int_0^\infty e^{-n(\sigma-j\beta)t}\, g(t)\, dt = \int_0^\infty e^{-st}\, g(t)\, dt\,\Big|_{s=s_n} = \mathcal{L}\{g(t)\}\Big|_{s=s_n} = G(s_n) \qquad n = 1,2,\ldots,M \tag{132}$$

Consequently, knowledge of the Laplace transform, $G(s_n)$, of $g(t)$ at the M complex frequencies $s_n = n(\sigma-j\beta)$, $n = 1,2,\ldots,M$, is sufficient to determine the expansion coefficients a_m of Eq. (131) and, thereby, the optimum integrated-squared-error approximation $\hat{g}(t)$ to the prescribed function $g(t)$ on $t > 0$. Alternatively, this algorithm for representing $g(t)$ in terms of $X_m(t)$ also provides a technique for obtaining an approximate numerical inverse $\hat{g}(t)$ to the prescribed Laplace transform, $G(s)$, of an unknown function $g(t)$.

Through Eqs. (130)-(132), the approximant $\hat{g}(t)$ of $g(t)$ can be expressed as

$$\hat{g}(t) = \sum_{m=1}^{M} \sum_{n=1}^{m} \sqrt{2\pi}\, \lambda_{mn}^*\, G(s_n)\, e^{-n(\sigma-j\beta)t} \qquad t > 0 \tag{133}$$

As a consequence of the Fourier transform relationship between corresponding elements $X_m(t)$ and $U_m(\omega)$, a least-integrated-squared-error representation $\hat{h}(\omega)$ to a specified function $h(\omega) \subset L^2(-\infty,\infty)$ can also be readily computed. If $h(\omega)$ is the Fourier transform of $g(t)$ and if $\hat{h}(\omega)$ is defined as

$$\hat{h}(\omega) \equiv \sum_{m=1}^{M} b_m U_m(\omega) \approx h(\omega) \tag{134}$$

with the coefficients b_m selected to yield

$$\min_{\{b_m\}} \| h(\omega) - \hat{h}(\omega) \|^2 \qquad m = 1,2,\ldots,M \tag{135}$$

then, following the solution for the Fourier coefficients a_m, the b_m are determined as

$$b_m = \int_{-\infty}^{\infty} h(\omega) U_m^*(\omega)\, d\omega \qquad m = 1,2,\ldots,M \tag{136}$$

But by Parseval's theorem and the previously established relation $\mathcal{F}\{X_m(t)\} = \sqrt{2\pi}\, U_m(\omega)$, it follows that

$$\frac{1}{2\pi} \int_{-\infty}^{\infty} h(\omega) U_m^*(\omega)\, d\omega = \frac{1}{\sqrt{2\pi}} \int_0^{\infty} g(t) X_m^*(t)\, dt \tag{137}$$

so that substitution of Eqs. (131) and (135) in Eq. (137) yields the numerical equivalence

$$b_m = \sqrt{2\pi}\, a_m \qquad m = 1,2,\ldots,M \tag{138}$$

This result and another application of Parseval's theorem can provide a link between the approximation errors incurred in the

domains $\{X_m(t)\}$, $g(t) \subset L^2(0,\infty)$, and $\{U_m(\omega)\}$, $h(\omega) \subset L^2(-\infty,\infty)$. If ϵ denotes the error in the representation $\hat{g}(t) \sim g(t)$; i.e.,

$$\epsilon = \min_{\{a_m\}} \| g(t) - \hat{g}(t) \|^2 = \min_{\{a_m\}} \int_0^\infty | g(t) - \hat{g}(t) |^2 \, dt \tag{139}$$

then from Eq. (130) for $\hat{g}(t)$, ϵ can be written as

$$\epsilon = \int_0^\infty \Bigg[g(t)\, g^*(t) - g^*(t) \sum_{m=1}^{M} a_m X_m(t) - g(t) \sum_{m=1}^{M} a_m^* X_m^*(t)$$

$$+ \sum_{m=1}^{M} \sum_{n=1}^{M} a_m a_n^* X_m(t)\, X_n^*(t) \Bigg] dt \tag{140}$$

or, taking account of Eq. (131) for the optimum a_m and Eq. (26) for the orthonormality of the $X_m(t)$

$$\epsilon = \int_0^\infty | g(t) |^2 \, dt - \sum_{m=1}^{M} a_m a_m^* - \sum_{m=1}^{M} a_m^* a_m + \sum_{m=1}^{M} a_m a_m^* \tag{141}$$

so that

$$\epsilon = \| g(t) \|^2 - \sum_{m=1}^{M} | a_m |^2 \tag{142}$$

In a similar fashion, the error, $\mathcal{E}$, in the representation $\hat{h}(\omega)$ of $h(\omega) \subset L^2(-\infty,\infty)$ can be derived from Eqs. (134)-(136) and (22) as

$$\mathcal{E} = \min_{\{b_m\}} \| h(\omega) - \hat{h}(\omega) \|^2 = \| h(\omega) \|^2 - \sum_{m=1}^{M} | b_m |^2 \tag{143}$$

From the Fourier transform relation of $g(t)$ to $h(\omega)$, Parseval's theorem guarantees that

$$|g(t)|^2 = \frac{1}{2\pi} |h(\omega)|^2 \tag{144}$$

Consequently, ϵ and $\hat{\mathcal{E}}$ are related as

$$\hat{\mathcal{E}} = 2\pi \; \|g(t)\|^2 - \sum_{m=1}^{M} 2\pi \, |a_m|^2 = 2\pi \, \epsilon \tag{145}$$

Since $|a_m|^2$ is nonnegative, Eq. (145) indicates that both ϵ and $\hat{\mathcal{E}}$ are monotonically nonincreasing as M is increased. Therefore, for an arbitrarily prescribed approximation error ϵ or $\hat{\mathcal{E}}$, M can be iteratively ascertained. Moreover, according to Eqs. (131) and (136) for a_m and b_m, M can be determined without recomputing any of the previously calculated coefficients a_m or b_m, $m = 1,2,\ldots,M-1$.

The key results of this section are recapitulated in Table 3. In the table, the inner product notation $(x(\zeta),y(\zeta))$ denotes $\int_{-r_1}^{r_2} x(\zeta)\, y^*(\zeta)\, d\zeta$, where the appropriate values of r_1 and r_2 are obvious from the context.

Table 3

REPRESENTATION RELATIONS

Property	Approximant		Relation
Form	$\hat{g}(t) = \sum_{m=1}^{M} a_m X_m(t)$	$\hat{h}(\omega) = \sum_{m=1}^{M} b_m U_m(\omega)$	$\hat{g}(t) = \sqrt{2\pi}\, \mathcal{F}^{-1} \{h(\omega)\}$
Admissible functions	$g(t) \subset L^2(0,\infty)$	$h(\omega) \subset L^2(-\infty,\infty)$	$g(t) = \mathcal{F}^{-1} \{h(\omega)\}$
Least-square coefficients	$a_m = (g(t), X_m(t))$	$b_m = (h(\omega), U_m(\omega))$	$a_m = b_m / \sqrt{2\pi}$
Least-square error	$\epsilon = \|g(t)\|^2 - \sum_{m=1}^{M} \|a_m\|^2$	$\mathcal{E} = \|h(\omega)\|^2 - \sum_{m=1}^{M} \|b_m\|^2$	$\epsilon = \mathcal{E}/2\pi$

VII. COMPUTATIONAL ASPECTS OF ORTHONORMAL EXPANSIONS IN $U_m(\omega)$ and $X_m(t)$

The computational labor in evaluating the coefficients λ_{mn} from Eq. (45) can be substantially reduced by employing a recursive procedure. In order to develop such a recursion scheme, the coefficients $\lambda_{m+1,n}$ must be expressed in terms of previously determined $\lambda_{m,n}$.[†] This can be accomplished by using Eq. (45).

With m replaced by m+1 in Eq. (45), the coefficient $\lambda_{m+1,n}$ becomes

$$\lambda_{m+1,n} = \begin{cases} \sqrt{\dfrac{(m+1)\sigma}{\pi}}\, \dfrac{\prod\limits_{r=1}^{m} \left[n+r\left(\dfrac{\sigma+j\beta}{\sigma-j\beta}\right)\right]}{\prod\limits_{r=1}^{m+1}{}' \, [r-n]} & \begin{array}{l} m = 1,2,\ldots \\ 1 \le n \le m+1 \end{array} \\ \\ 0 & \begin{array}{l} m = 1,2,\ldots \\ n > m+1 \end{array} \end{cases} \tag{146}$$

Since λ_{mn} is also determined by Eq. (45), the ratio of $\lambda_{m+1,n}$ to λ_{mn} can be formed as

$$\frac{\lambda_{m+1,n}}{\lambda_{m,n}} = \begin{cases} \sqrt{2}\,(1+z) & \begin{array}{l} m = 1 \\ n = 1 \end{array} \\ \\ \sqrt{\dfrac{m+1}{m}}\, \dfrac{\prod\limits_{r=1}^{m} (n+rz)}{\prod\limits_{r=1}^{m+1}{}' \, (r-n)} \, \dfrac{\prod\limits_{r=1}^{m}{}' \, (r-n)}{\prod\limits_{r=1}^{m-1} (n+rz)} & \begin{array}{l} m = 2,3,\ldots \\ 1 \le n \le m \end{array} \end{cases} \tag{147}$$

[†] In order to avoid confusion, a comma is used to separate the subscripts m and n in the coefficients λ_{mn}. Accordingly, λ_{mn} is understood to signify $\lambda_{m,n}$, the n^{th} coefficient of the m^{th} basis function.

Consequently,

$$\lambda_{m+1,n} = \sqrt{\frac{m+1}{m}} \; \frac{n(\sigma - j\beta) + m(\sigma + j\beta)}{(m+1-n)\;(\sigma - j\beta)} \; \lambda_{m,n} \qquad \begin{array}{l} m = 1,2,\ldots \\ 1 \le n \le m \end{array} \qquad (148)$$

Since λ_{mn} is zero for $n > m$, and since division by zero is invalid, the above expression for $\lambda_{m+1,n}$ in terms of the preceding $\lambda_{m,n}$ is limited to the range $1 \le n \le m$. In order to determine the remaining element $\lambda_{m+1,m+1}$, the identity given by Eq. (110) can be used in the following way

$$\sum_{n=1}^{m} \frac{\lambda_{mn}}{n} = \sqrt{\frac{\sigma}{m\pi}} \left[\frac{\sigma + j\beta}{\sigma - j\beta}\right]^{m-1} \qquad m = 1,2,\ldots \qquad (110)$$

With m replaced by m+1, Eq. (110) becomes

$$\sum_{n=1}^{m+1} \frac{\lambda_{m+1,n}}{n} = \sqrt{\frac{\sigma}{(m+1)\pi}} \left[\frac{\sigma + j\beta}{\sigma - j\beta}\right]^{m} \qquad (149)$$

If the term $n = m+1$ is individually summed, Eq. (149) gives

$$\sum_{n=1}^{m} \frac{\lambda_{m+1,n}}{n} + \frac{1}{m+1} \lambda_{m+1,m+1} = \sqrt{\frac{\sigma}{(m+1)\pi}} \left[\frac{\sigma + j\beta}{\sigma - j\beta}\right]^{m} \qquad (150)$$

so that

$$\lambda_{m+1,m+1} = \sqrt{\frac{(m+1)\sigma}{\pi}} \left[\frac{\sigma + j\beta}{\sigma - j\beta}\right]^{m} - \sum_{n=1}^{m} \frac{m+1}{n} \lambda_{m+1,n} \qquad m = 1,2,\ldots \qquad (151)$$

With this last relation and Eq. (148), it is evident that the coefficients $\lambda_{m,n}$ can be recursively evaluated by using the initial values given in Eq. (45), namely

$$\begin{array}{lll} \lambda_{11} = \sqrt{\dfrac{\sigma}{\pi}} & \begin{array}{l} m = 1 \\ n = 1 \end{array} & \\ & & (152) \\ \lambda_{1n} = 0 & n > 1 & \end{array}$$

and by employing the expressions

$$\lambda_{m+1,n} = \begin{cases} \sqrt{\frac{m+1}{m}}\,\frac{n(\sigma-j\beta)+m(\sigma+j\beta)}{(m+1-n)\;(\sigma-j\beta)}\,\lambda_{m,n} & m = 1,2,\ldots \quad 1 \leq n \leq m \\ \sqrt{\frac{(m+1)\sigma}{\pi}}\left[\frac{\sigma+j\beta}{\sigma-j\beta}\right]^{m} - (m+1)\sum_{k=1}^{m}\frac{\lambda_{m+1,k}}{k} & m = 1,2,\ldots \quad n = m+1 \\ 0 & m = 1,2,\ldots \quad n > m+1 \end{cases} \tag{153}$$

The check-sum relation Eq. (67) still applies and can be used to detect errors in the evaluation of each new row of basis coefficients $\lambda_{m+1,n}$ ($n = 1,2,\ldots, m+1$; $m = 1,2,\ldots$) generated from the previously computed $\lambda_{m,n}$.

It is also possible to provide a recursion formula for the efficient computation of the orthonormal elements $\{X_m(t)\}$ and $\{U_m(\omega)\}$. From Eqs. (3) and (18), $X_m(t)$ can be expressed as the sum of exponentials

$$X_m(t) = \sqrt{2\pi}\sum_{n=1}^{m}\lambda_{mn}\,e^{-n(\sigma-j\beta)t} \qquad m = 1,2,\ldots \tag{154}$$

The fact that this sum involves integrally related decay factors can be exploited to evaluate $X_m(t)$ recursively for arbitrary values of $t > 0$. This is accomplished by first defining the associated quantities $X_{m,r}$ as

$$X_{m,1} = \lambda_{mm} \qquad r = 1 \tag{155}$$

with

$$\lambda_{m0} = 0 \tag{156}$$

and with λ_{mn}, $n \neq 0$, given by the iterative relation Eq. (153). With these definitions, it is clear that m iterations of the expression

$$X_{m,r+1} = e^{-(\sigma-j\beta)t} X_{m,r} + \lambda_{m,m-r} \qquad \begin{array}{l} r = 1,2,\ldots,m \\ m = 1,2,\ldots \end{array} \tag{157}$$

and final multiplication by $\sqrt{2\pi}$ yields the m^{th} basis function

$$X_m(t) = \sqrt{2\pi}\, X_{m,m+1}(t) \tag{158}$$

Thus, for arbitrary t, $X_m(t)$ can be calculated with only one evaluation of $e^{-(\sigma-j\beta)t}$, with m+1 multiplications, and with m-1 additions. This is an important economy in either manual or machine computation time as it results in only one exponential table look-up or one exponential subroutine entry.

By following the calculation of $X_m(t)$, $U_m(\omega)$ can also be iteratively obtained for any value of ω. From the rational function form, Eq. (41), of $U_m(\omega)$, it follows that

$$U_{m+1}(\omega) = \sqrt{\frac{(m+1)\sigma}{\pi}} \; \frac{\prod_{n=1}^{m} [n\sigma - j(\omega - n\beta)]}{\prod_{n=1}^{m+1} [n\sigma + j(\omega - n\beta)]} \qquad m = 1,2,\ldots \tag{159}$$

Consequently,

$$U_{m+1}(\omega) = \sqrt{\frac{m+1}{m}} \; \frac{[m\sigma - j(\omega - m\beta)]}{[(m+1)\sigma + j[\omega - (m+1)\beta]]} \, U_m(\omega) \qquad m = 1,2,\ldots \tag{160}$$

with

$$U_1(\omega) = \sqrt{\frac{\sigma}{\pi}} \; \frac{1}{\sigma + j(\omega - \beta)} \tag{161}$$

Once $U_1(\omega)$ is calculated for an arbitrary value of ω, then $U_2(\omega)$, $U_3(\omega), \ldots,$ follow by successive multiplications by $\sqrt{2}(\sigma-j\omega+j\beta)/(2\sigma+j\omega-2j\beta)$, $\sqrt{3/2}\,(2\sigma-j\omega+2j\beta)/(3\sigma+j\omega-3j\beta)$, and so forth.

It has already been noted that computation of the expansion coefficients for a least-integrated-squared-error representation of a prescribed g(t) requires knowledge of the Laplace transform $G(s_n)$ of g(t) at the M complex frequencies $s_n = n(\sigma+j\beta)$, $n = 1,2,\ldots,M$. Thus, evaluation of the coefficients a_m in Eq. (131) entails integrals of the type

$$L_n\{g(t)\} \equiv G(s_n) = \int_0^\infty g(t)\ e^{-n(\sigma+j\beta)t}\,dt \qquad n = 1,2,\ldots,M \qquad (162)$$

When the Laplace transform of g(t) is not available, it can be approximately determined by a variety of quadrature schemes.[15]

If the linear functionals L_n are approximated by the quadrature formula

$$L_n \approx \sum_{k=1}^{M} w_k\, g(t_k) \qquad n = 1,2,\ldots,M \qquad (163)$$

then an appropriate criterion for obtaining the weights w_k, $k = 1,2,\ldots,M$ for assigned sample points $t_1,t_2,\ldots,t_M$ is that L_n be exact for $g(t) = 1$, $e^{-(\sigma+j\beta)t}$, $e^{-2(\sigma+j\beta)t}, \ldots, e^{-(M-1)(\sigma+j\beta)t}$. This criterion translates into the system of equations

$$\left.\begin{aligned}
\sum_{k=1}^{M} w_k &= \int_0^\infty e^{-n(\sigma+j\beta)t}\,dt = \frac{1}{n(\sigma+j\beta)} \\
\sum_{k=1}^{M} w_k e^{-(\sigma+j\beta)t_k} &= \int_0^\infty e^{-(n+1)(\sigma+j\beta)t}\,dt = \frac{1}{(n+1)(\sigma+j\beta)} \\
&\vdots \\
\sum_{k=1}^{M} w_k e^{-(M-1)(\sigma+j\beta)t_k} &= \int_0^\infty e^{-(n+M-1)(\sigma+j\beta)t}\,dt = \frac{1}{(n+M-1)(\sigma+j\beta)}
\end{aligned}\right\} \tag{164}$$

It is convenient to employ the following matrix notations in the above equations:

$$\underline{w} = \begin{bmatrix} w_1 \\ w_2 \\ \vdots \\ w_M \end{bmatrix} \tag{165}$$

$$\underline{V} = \begin{bmatrix} 1 & 1 & \cdots & 1 \\ e^{-(\sigma+j\beta)t_1} & e^{-(\sigma+j\beta)t_2} & \cdots & e^{-(\sigma+j\beta)t_M} \\ \vdots & \vdots & & \vdots \\ e^{-(M-1)(\sigma+j\beta)t_1} & e^{-(M-1)(\sigma+j\beta)t_2} & \cdots & e^{-(M-1)(\sigma+j\beta)t_M} \end{bmatrix} \tag{166}$$

$$\underline{m} = \begin{bmatrix} \frac{1}{n(\sigma+j\beta)} \\ \frac{1}{(n+1)(\sigma+j\beta)} \\ \cdot \\ \cdot \\ \cdot \\ \frac{1}{(n+M-1)(\sigma+j\beta)} \end{bmatrix} \tag{167}$$

With these matrix definitions, the system of equations in Eq. (164) can be written as

$$\underline{V}\,\underline{w} = \underline{m} \tag{168}$$

and the symbolic solution for the weights becomes

$$\underline{w} = \underline{V}^{-1}\,\underline{m} \tag{169}$$

$\underline{V}$ is identifiable as the Vandermonde matrix and $\underline{V}^{-1}$ as its inverse.[†] Since σ must be nonzero and positive, and since the sample points t_i, $i = 1,2,\ldots,M$ are distinct, $\underline{V}$ is nonsingular[††] and has an inverse, $\underline{V}^{-1}$. Once $\underline{V}^{-1}$ is evaluated, the solution for the quadrature weights can be obtained from Eq. (169) and the expansion coefficients a_m can be finally computed.

Numerous expressions are available for determining the elements of the inverse Vandermonde matrix.[16-18] With a judicious choice

† See Ref. 10, p. 92.

†† See Ref. 10, p. 93.

of the sampling points, t_i, the inverse $\underline{V}^{-1}$ can be computed readily either from greatly simplified formulas† or written directly from tables of Universal matrices.(19)†† An application of Universal matrices to a quadrature scheme similar to Eq. (163) can be found in Ref. 1.††† Because of the direct analogy to the present situation (with σ replaced by $(\sigma+j\beta)$), the derivations will not be repeated here. By using the Universal matrices and the special sampling points t_i, it is possible to derive the weights $\underline{w}$ from a simple computation of the moment vector $\underline{m}$ and a premultiplication by the known matrix $\underline{V}^{-1}$. The expansion coefficients a_m then follow from Eqs. (163) and (131). Since a_m and b_m are related through Eq. (138), a similar development can be made for efficiently determining the coefficients b_m in the approximation $\hat{h}(\omega)$ to a prescribed function $h(\omega)$.

---

†See Ref. 16, pp. 96-98.

††Universal matrices are the inverses of the Vandermonde matrices with sample points t_i = i-n/2, i=0,1,...,n and with n equal to an integer.

†††See Ref. 1, pp. 55-61.

VIII. SELECTION OF THE COMPLEX DECAY CONSTANT (σ-jβ)

The orthonormal functions $X_m(t)$ and $U_m(\omega)$, the generating coefficients λ_{mn} and γ_{mn}, and the expansion coefficients a_m and b_m discussed in Section VII are dependent on σ and β. Except for the constraints that σ and β be real and that σ be greater than zero, the parameters σ and β have not yet been specified.

In attempting to formulate a criterion for selecting σ and β, several difficulties are immediately encountered. First, in data which arise from a sum of exponentially damped sinusoids of unknown decay and frequency constants, it may not be possible to obtain a unique complex decay value (or pole location) such that $-n(\sigma-j\beta)$, $n = 1,2,\ldots$, matches all the decay factors inherent in some prescribed data $g(t)$, or that matches all the poles comprised by a given $h(\omega)$. Second, there always exist functions $g(t)$ or $h(\omega)$ for which the optimum choice of σ and β in an M-term representation does not remain best as the approximation complexity is increased.

Another problem in solving for the complex decay constant (σ-jβ) occurs when the given data is impaired by noise or when the subsequent manipulation of the data is accompanied by round-off errors. The intractability of this classical problem is widely recognized and has been amply illustrated even in cases where four or fewer exponentials underlie the numerical data.[(20-30)†] Consequently, the present objective in solving for σ and β will not be to recover the original parameters imbedded in given data. Instead, an approximation to the

† See Ref. 15, pp. 272-288.

data will be sought which is best in an integral-square sense with respect to the Fourier coefficients a_m or b_m. Another more tractable criterion may be necessary to optimize the approximation with respect to σ and β.

Since σ and β are subject to only the two aforementioned constraints, their choice is essentially arbitrary and can be based on any criterion of optimality. One obvious criterion is the minimization over σ and β of the integral-square approximation error ϵ or $\mathcal{E}$ given by Eqs. (142) and (143). Although this objective would be consistent with the conditions leading to the expansion coefficients a_m and b_m, it unfortunately results in a nonlinear programming problem requiring iterative search procedures for its solution.[†]

A more tractable criterion than least-squares is one which requires matching the asymptotic approach to zero of the approximant and prescribed function for large t. More precisely, if $g(t)$ and its derivative $g'(t)$ exist for some $t \gg 1$ and are nonzero, both σ and β can be determined by requiring $g(t)$ and its approximant $\hat{g}(t)$ to have the same decay envelope for large t. This condition can be derived from Eqs. (130), (18), and (3), as follows.

Since

$$g(t) \approx \hat{g}(t) = \sum_{m=1}^{M} a_m X_m(t) = \sum_{m=1}^{M} a_m \sum_{n=1}^{m} \sqrt{2\pi}\, \lambda_{mn}\, e^{-n(\sigma-j\beta)t} \qquad t > 0 \qquad (170)$$

[†]See Ref. 1, p. 40.

the first derivative of g(t) can be written as

$$g'(t) \approx \hat{g}'(t) = -\sqrt{2\pi}\,(\sigma-j\beta) \sum_{m=1}^{M} a_m \sum_{n=1}^{m} n\,\lambda_{mn}\, e^{-n(\sigma-j\beta)t} \qquad t>0 \qquad (171)$$

For sufficiently large t and with $\sigma > 0$, it is clear that only the terms with the smallest decay factor, $-\sigma$, predominate in Eqs. (170) and (171). Consequently, Eqs. (170) and (171) can be simplified for $t \gg 1$ to

$$g(t) \approx \hat{g}(t) \approx \sqrt{2\pi}\, e^{-(\sigma-j\beta)t} \sum_{m=1}^{M} a_m \lambda_{m1} \qquad t\gg 1 \qquad (172)$$

and

$$g'(t) \approx \hat{g}'(t) \approx -\sqrt{2\pi}\,(\sigma-j\beta)\, e^{-(\sigma-j\beta)t} \sum_{m=1}^{M} a_m \lambda_{m1} \qquad t\gg 1 \qquad (173)$$

If the ratio of these last two equations is formed, then for $\sigma > 0$

$$-\left.\frac{g'(t)}{g(t)}\right|_{t\gg 1} \approx -\left.\frac{\hat{g}'(t)}{\hat{g}(t)}\right|_{t\gg 1} \approx \sigma-j\beta \qquad \left.g(t)\right|_{t\gg 1} \neq 0 \qquad (174)$$

Hence,

$$\sigma \approx -\mathrm{Re}\left[\left.\frac{g'(t)}{g(t)}\right|_{t\gg 1}\right] \qquad g(t) \neq 0, \quad g'(t) \neq 0 \qquad (175a)$$

and

$$\beta \approx \mathrm{Im}\left[\left.\frac{g'(t)}{g(t)}\right|_{t\gg 1}\right] \qquad (175b)$$

When the prescribed function g(t) is real, the solution for β given by Eq. (175b) indicates that β can be set equal to zero. However,

this may be an unsatisfactory choice for β if $g(t)$ exhibits an oscillatory nature for some values of t. In such cases, β and σ can be resolved by representing $g(t)$ not as in Eq. (170), but as

$$g(t) \approx \hat{g}(t) = \sum_{m=1}^{M} c_m Y_m(t) = \frac{1}{\sqrt{2}} \sum_{m=1}^{M} c_m \left[X_m(t) + X_m^*(-t)\right] \tag{176}$$

where the orthonormal functions $Y_m(t)$ are defined by Eqs. (55), (61) and (62). In view of Eq. (62), $\hat{g}(t)$ becomes

$$g(t) \approx \sqrt{\pi} \sum_{m=1}^{M} c_m \sum_{n=1}^{m} |\lambda_{mn}| \, e^{-n\sigma|t|} \left\{ \cos\left[n\beta t + \frac{t}{|t|} \arg(\lambda_{mn})\right] + \frac{t}{|t|}\, j \sin[n\beta|t| + \arg(\lambda_{mn})] \right\} \qquad |t| < \infty \tag{177}$$

For $t \gg 1$, the terms for which $n = 1$ predominate, as before, so that

$$g(t) \approx \sqrt{\pi}\, e^{-\sigma t} \sum_{m=1}^{M} c_m |\lambda_{m1}| \left\{ \cos[\beta t + \arg(\lambda_{m1})] + j \sin[\beta t + \arg(\lambda_{m1})] \right\} \qquad t \gg 1 \tag{178}$$

with

$$g'(t) \approx -\sigma\, g(t) + \sqrt{\pi}\, e^{-\sigma t} \sum_{m=1}^{M} c_m |\lambda_{m1}| \left\{ -\beta \sin[\beta t + \arg(\lambda_{m1})] + j\beta \cos[\beta t + \arg(\lambda_{m1})] \right\} \qquad t \gg 1 \tag{179}$$

and

$$g''(t) \approx -\sigma\, g'(t) + \sqrt{\pi}\, e^{-\sigma t}\, \sigma\beta \sum_{m=1}^{M} c_m |\lambda_{m1}| \Bigg\{ \sin[\beta t + \arg(\lambda_{m1})]$$

$$- j \cos[\beta t + \arg(\lambda_{m1})] \Bigg\} - \sqrt{\pi}\, e^{-\sigma t}\, \beta^2 \sum_{m=1}^{M} c_m |\lambda_{m1}| \Bigg\{ \cos[\beta t + \arg(\lambda_{m1})]$$

$$+ j \sin[\beta t + \arg(\lambda_{m1})] \Bigg\} \qquad t \gg 1 \qquad (180)$$

or

$$g''(t) \approx -2\sigma g'(t) - (\sigma^2+\beta^2)\, g(t) \qquad t \gg 1 \qquad (181)$$

Similarly,

$$g'''(t) \approx -2\sigma\, g''(t) - (\sigma^2+\beta^2)\, g'(t) \qquad t \gg 1 \qquad (182)$$

The preceding two equations relate the unknowns σ and β to the given function $g(t)$ and its derivatives at some large value of t. Equation (182) can be solved for β^2 to yield

$$\beta^2 \approx - \left. \frac{g'''(t) + 2\sigma g''(t) + \sigma^2 g'(t)}{g'(t)} \right|_{t \gg 1} \qquad g'(t)\Big|_{t \gg 1} \neq 0 \qquad (183)$$

When this expression for β^2 is substituted into Eq. (181), the following relation for σ is obtained

$$\sigma \approx \left. \frac{g'''(t)g(t) - g'(t)g''(t)}{2[g'^2(t) - g(t)g''(t)]} \right|_{t \gg 1} \qquad g'(t) \neq \pm\sqrt{g(t)g''(t)},\ t \gg 1 \qquad (184)$$

Thus, β is also explicitly related to $g(t)$ and its derivatives for some large value of t as

$$\beta \approx \left\{ -\frac{g'''(t)}{g'(t)} - \frac{g''(t)}{g'(t)} \cdot \frac{g'''(t)g(t)-g'(t)g''(t)}{g'^2(t) - g(t)g''(t)} - \frac{g'''^2(t)g^2(t)+g''^2(t)g'^2(t)-2g(t)g'(t)g''(t)g'''(t)}{4[g'^4(t)+g^2(t)g''^2(t)-2g'^2(t)g(t)g''(t)]} \right\}^{\frac{1}{2}} \Bigg|_{t\gg1} \qquad (185)$$

$$g'(t) \neq 0, \; g'(t) \neq \pm g(t)g''(t), \qquad t\gg1$$

Several qualifications are noteworthy regarding these equations for σ and β. It is clear that Eqs. (175), or Eqs. (184) and (185), provide suitable values of $(\sigma-j\beta)$ when the prescribed g(t) is given as a long-time record. When g(t) is a transient or pulse-type function, or when the complete time history of g(t) is unknown, the asymptotic properties of g(t) are not available for estimating σ and β by the aforementioned equations. For these pulsatile functions, it is necessary to resort to more elaborate techniques for obtaining $(\sigma-j\beta)$.

One procedure for obtaining σ and β relies on the discrete version of Prony's exponential approximation method.† In essence, an application of the Prony algorithm[31] enables a selection of $(\sigma-j\beta)$ based on many samples of g(t), rather than on a match of only the asymptotic values of g(t) and its derivatives for large t. Other versions of Prony's scheme[32-33] which are available can also be used to select σ and β. Depending on the procedure chosen, the approximant $\hat{g}(t)$ satisfies in a least-squares sense a finite difference or differential equation involving the sampled or continuous ordinates of the specified g(t). Since all these approaches are conceptually similar, and since

---

†See Ref. 1, pp. 42-51.

the discrete version of Prony's method has been detailed in Ref. 1 for the case in which $\beta = 0$, the derivations will not be repeated here with $\beta \neq 0$. It will suffice to state that the derivations in Ref. 1 apply to the condition $\beta \neq 0$ considered in this Memorandum by merely substituting $(\sigma - j\beta)$ for σ in Eqs. (127)-(154) of Ref. 1.

Though no mention has yet been made of the associated problem of selecting σ and β for approximations of functions $h(\omega) \subset L^2(-\infty,\infty)$ by sums of the orthonormal elements $U_m(\omega)$, it turns out that all the techniques discussed so far are applicable, albeit indirectly and with additional computational labor. Since the approximants $\hat{h}(\omega)$ and $\hat{g}(t)$ are Fourier transform pairs, the Fourier transform can be taken of a graphically or analytically specified $h(\omega)$ to produce a $g(t)$ or samples of $g(t)$ at S points $t = 0,1,\ldots,S-1$.[†] In turn, these ordinates of $g(t)$ can be utilized in all of the Prony schemes cited earlier, and a choice of $(\sigma - j\beta)$ can again be made based on $g(t)$'s satisfying a difference or differential equation.

[†] See Ref. 30, pp. 67-75.

IX. NUMERICAL EXAMPLES AND APPLICATIONS TO FILTER DESIGN

In the introduction, several important practical applications are enumerated for the orthonormal sets $X_m(t)$, $Y_m(t)$, $U_m(\omega)$, and $V_m(\omega)$. In order to clarify the use of the algorithms developed in this Memorandum, two filter design problems will be illustrated in this section.

The first example pertains to an approximation problem in network synthesis.[34] It requires finding a physically realizable transfer function for a finite, lumped-element, passive, linear network such that the network's impulse response is a replica of the waveform depicted in Fig. 1. The approximating response must be close enough to the prescribed response that the mean-square error between them is less than $\frac{1}{8} \times 10^{-2}$ over the time interval (0,10). In quantitative terms, the prescribed transient response is given as

$$g(t) = \begin{cases} 0 & t < 0 \\ 2t & 0 \leq t \leq 0.5 \\ -2(t-1) & 0.5 \leq t \leq 0.95 \\ e^{-2.42377t} & t > 0.95 \end{cases} \qquad (186)$$

and it is necessary to find a_k and N for the approximant $\hat{g}(t)$ such that

$$\hat{g}(t) = \sum_{k=1}^{N} a_k X_k(t) = \sum_{k=1}^{N} a_k \sqrt{2\pi} \sum_{m=1}^{k} \lambda_{km} e^{-m(\sigma-j\beta)t} \approx g(t) \qquad (187)$$

and

$$\epsilon_N = \frac{1}{10} \int_0^{10} |g(t)-\hat{g}(t)|^2 dt \leq \delta = \frac{1}{8} \times 10^{-2} \qquad (188)$$

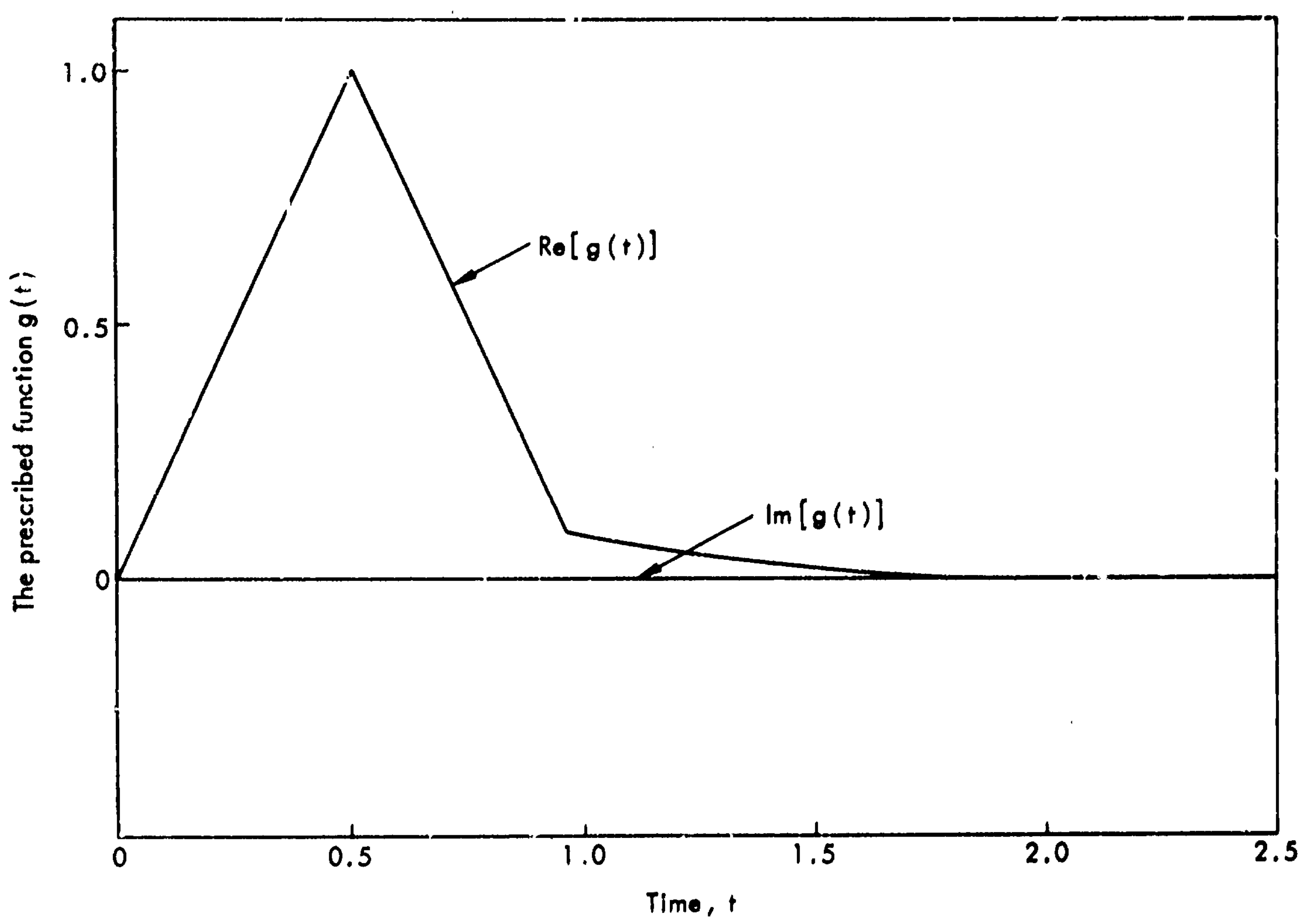

Fig. 1 — Graph of prescribed $g(t)$

In order to meet this design specification, the coefficients a_k, $k = 1,2,\ldots,N$ must be computed for some choice of σ, β, and N. In so doing it is convenient to recall that the expansion coefficients a_k are independent of N. Thus, N can be selected arbitrarily, ϵ_N can be evaluated and compared with δ, and N can be iteratively adjusted to be smaller or larger depending on whether $\epsilon_N < \delta$ or $\epsilon_N > \delta$. In either case, the a_k are determined from Eq. (131) as

$$a_k = \int_0^\infty g(t)\, X_m^*(t)\, dt = \sqrt{2\pi} \sum_{m=1}^{k} \lambda_{km}^* \int_0^\infty g(t)\, e^{-m(\sigma+j\beta)t}\, dt \qquad (189)$$

$$k = 1,2,\ldots,N$$

Thus, evaluation of all the a_k entails integrating the N quantities

$$G_m = \int_0^\infty g(t)\, e^{-m(\sigma+j\beta)t}\, dt \qquad m = 1,2,\ldots,N \qquad (190)$$

or

$$G_m = 2\int_0^{0.5} t\, e^{-m(\sigma+j\beta)t}\, dt + 2\int_{0.5}^{0.95} (1-t)\, e^{-m(\sigma+j\beta)t}\, dt + \int_{0.95}^{\infty} e^{-(m\sigma+mj\beta+2.42377)t}\, dt \qquad (191)$$

$$= \frac{1}{m^2(\sigma+j\beta)^2}\left[-4e^{-0.5m(\sigma+j\beta)} + [2 - .1(\sigma+j\beta)m]\, e^{-0.95m(\sigma+j\beta)} + 2\right] \qquad (192)$$

$$m = 1,2,\ldots,N$$

In order to determine the moments G_m (the sampled Laplace transform of g(t)) by this last relation, a selection of σ and β is necessary. Since g(t) is a pulse-like function with an exponential

tail and no oscillatory character, Eq. (175) is appropriate for determining σ and β. Accordingly

$$\sigma \approx -\mathrm{Re}\left[\frac{g'(t)}{g(t)}\bigg|_{t>>1}\right] = -\mathrm{Re}\left[\frac{-2.42377e^{-2.42377t}}{e^{-2.42377t}}\bigg|_{t>>1}\right] \tag{193}$$

or

$$\sigma \approx 2.42377 > 0 \tag{194}$$

and

$$\beta \approx \mathrm{Im}\left[\frac{g'(t)}{g(t)}\bigg|_{t>>1}\right] = 0 \tag{195}$$

With these choices for σ and β, the first nine coefficients† a_k are found from Eqs. (192), (189), and the tabulated λ_{mn} of Appendix A as

$$\begin{aligned}
a_1 &= 0.3738 + j\,0. & a_5 &= -0.1227 + j\,0. \\
a_2 &= 0.3957 + j\,0. & a_6 &= -0.0412 + j\,0. \\
a_3 &= 0.0694 + j\,0. & a_7 &= 0.0284 + j\,0. \\
a_4 &= -0.1164 + j\,0. & a_8 &= 0.0441 + j\,0. \\
& & a_9 &= 0.0227 + j\,0.
\end{aligned} \tag{196}$$

These values for a_k can be inserted in Eq. (187) along with the tabulated λ_{mn} to find $\hat{g}(t)$ for $t > 0$. When this is done, the rms error over the interval (0,10) can be computed from Eq. (188) to give

$$\epsilon_9 = 0.2 \times 10^{-2} = \delta \qquad N = 9 \tag{197}$$

† The values of a_k listed in Eq. (196) have been rounded to four significant digits.

Since $\epsilon_8 = 0.3 \times 10^{-2} > \delta$, it is clear that the solution for N = 9, $\sigma = 2.42377$, $\beta = 0$, satisfies the design specification. For these values of N, σ, and β, graphs of g(t) and $\hat{g}(t)$ can be compared (see Figs. 2 and 3).

Finally, since the Laplace transform of $X_m(t)$ is

$$\int_0^\infty X_m(t)\, e^{-st}\, dt = \sum_{n=1}^{m} \sqrt{2\pi}\, \lambda_{mn} \int_0^\infty e^{(-n\sigma - s + jn\beta)t}\, dt$$

$$= \sqrt{2\pi} \sum_{n=1}^{m} \lambda_{mn} \frac{1}{s+n(\sigma-j\beta)} \qquad \left\{ \begin{array}{l} s = \alpha + j\beta \\ \alpha + n\sigma > 0 \\ n = 1,2,\ldots \end{array} \right. \tag{198a}$$

the approximating realizable network transfer function can be written as

$$\hat{H}(s) = \sqrt{2\pi} \sum_{m=1}^{N} \sum_{n=1}^{m} \lambda_{mn} \frac{a_m}{s+n(\sigma-j\beta)} \approx H(s) \equiv \int_0^\infty g(t)\, e^{-st}\, dt \tag{198b}$$

Thus, with σ equal to a real number and β equal to zero (as in the present example) $\hat{H}(s)$, the network approximant, can be implemented with either RC or RL elements to give the impulse response $\hat{g}(t) \approx g(t)$.

The second numerical example deals with a problem of optimal filtering. In processing signals impaired by additive noise, it is possible to accomplish smoothing and prediction by constructing an appropriate filter.(35) Usually this requires approximating the spectral densities of the signal and random noise process by rational functions. This procedure in turn permits the analytically optimum filter to be approximated in the form of a linear, lumped-constant network. Since the power spectral density is a rational function of frequency for a signal whose correlation function is a sum of

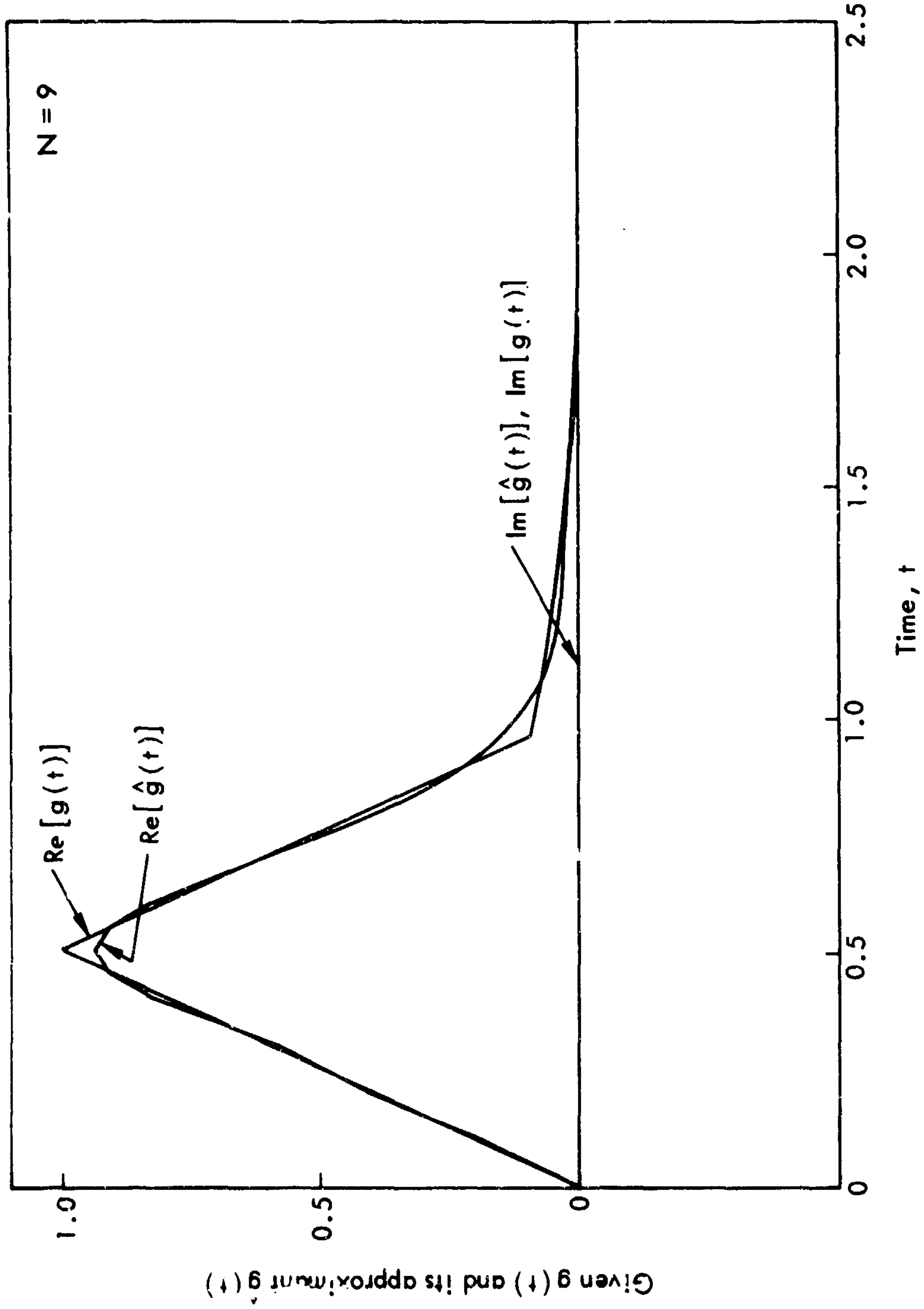

Fig. 2 — Graphs of g(t) and ĝ(t) for N = 9

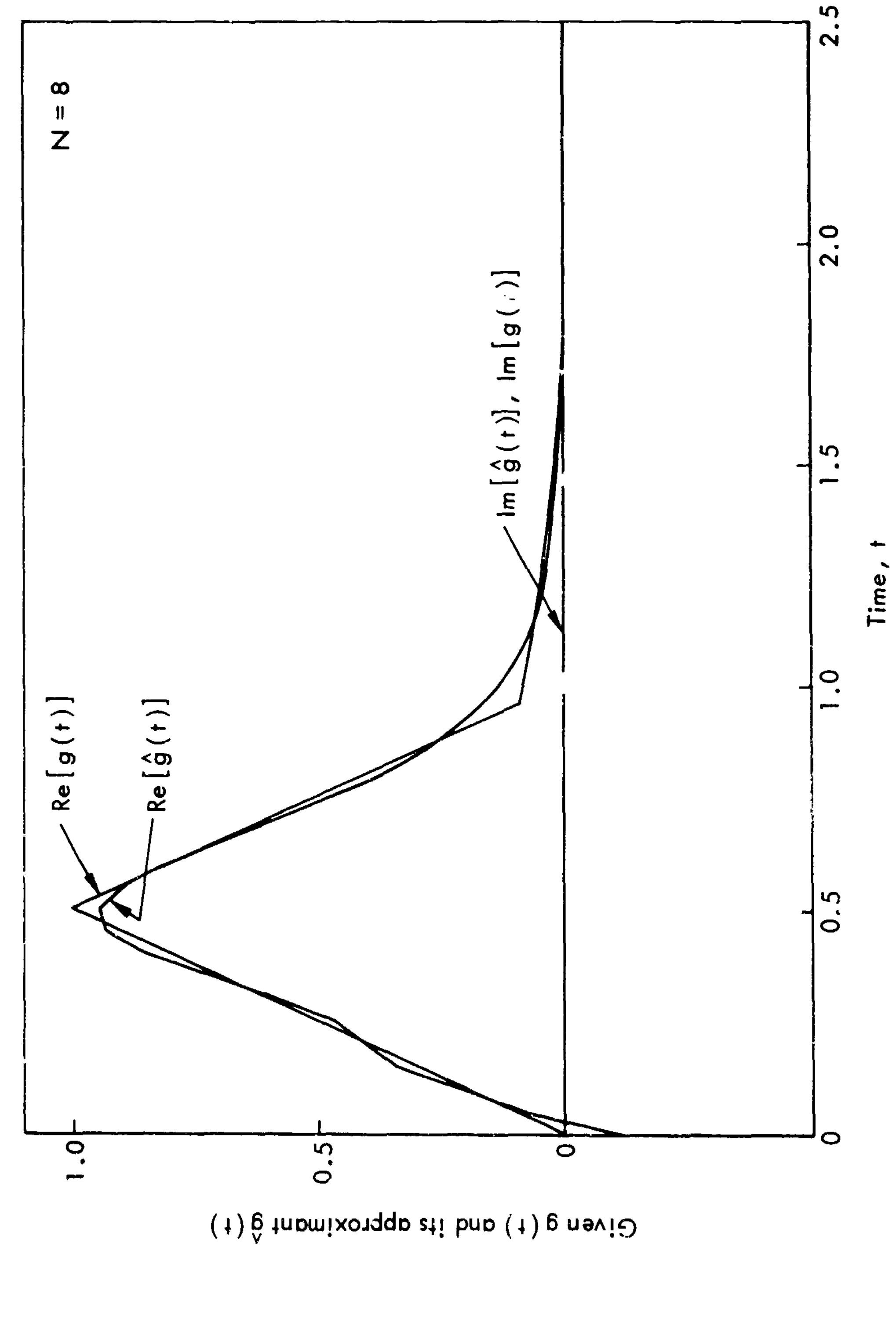

Fig.3 — Graphs of g(t) and ĝ(t) for N = 8

exponentials, it is possible to use advantageously the sets $\{X_m(t)\}$ and $\{Y_m(t)\}$ in the approximation of empirical correlation data.

A commonly observed autocorrelation function for a stationary random process is the exponentially damped cosine[36]

$$\Psi(\tau) = e^{-2|\tau|} \cos \pi\tau \qquad |\tau| > 0 \tag{199}$$

Though the best approximant of $\Psi(\tau)$ of exponential form is clearly $\Psi(\tau)$ itself, it is instructive to see how efficiently this exponentially damped cosinusoid can be approximated in an integral-square sense by the orthonormal elements $X_m(\tau)$. Since $\Psi(\tau)$ is defined over the doubly infinite interval, $|\tau| < \infty$, and since the $X_m(\tau)$ are non-zero only over $\tau > 0$, it is necessary to represent $\Psi(\tau)$ over $(-\infty,\infty)$ as

$$\Psi(\tau) \approx \hat{\Psi}_+(\tau) = \sum_{k=1}^{N} a_k X_k(\tau) \qquad 0 \le \tau < \infty \tag{200}$$

and

$$\Psi(\tau) \approx \hat{\Psi}_-(\tau) = \sum_{k=1}^{N} b_k X_k(\tau) \qquad -\infty < \tau \le 0 \tag{201}$$

It is evident from Eq. (199) that $\Psi(\tau)$ is an even function of τ; i.e.,

$$\Psi(\tau) = \Psi(-\tau) \tag{202}$$

Hence

$$\hat{\Psi}_+(\tau) = \hat{\Psi}_-(-\tau) \tag{203}$$

and

$$a_k = b_k \qquad k = 1,2,\ldots,N \tag{204}$$

Consequently, the moments G_m for $\Psi(\tau)$ can be related simultaneously to a_k and b_k through Eqs. (189), (190), and (204) as

$$G_m = \int_0^\infty e^{-(m\sigma+jm\beta+2)\tau} \cos \pi\tau \, d\tau \tag{205}$$

$$= \frac{m(\sigma+j\beta) + 2}{[m(\sigma+j\beta)+2]^2 + \pi^2} \qquad m = 1,2,\ldots,N$$

In terms of these G_m, the expansion coefficients for $\hat{\Psi}_+(\tau)$ and $\hat{\Psi}_-(\tau)$ become

$$a_k = \sqrt{2\pi} \sum_{m=1}^{k} \lambda_{km}^* G_m = b_k \qquad k = 1,2,\ldots,N \tag{206}$$

where the λ_{km}^* are tabulated in Appendix A as rational functions of σ and β.

In order to complete the approximation $\hat{\Psi}(\tau) \approx \Psi(\tau)$, a selection of the parameters σ and β is necessary. This will enable the evaluation of G_m, a_k, and b_k of Eqs. (205) and (206) and, thereby, of $\hat{\Psi}_+(\tau)$ and $\hat{\Psi}_-(\tau)$ in Eqs. (200) and (201). Since $\Psi(\tau)$ is given as a real, oscillatory, exponentially decaying function, Eqs. (183)-(185) can be used to obtain σ and β, with β not necessarily zero (as would be the case if Eq. (175) were used). Accordingly, the first three derivatives of $\Psi(\tau)$ for $\tau > 0$ become

$$\Psi'(\tau) = -e^{-2\tau} [2 \cos \pi\tau + \pi \sin \pi\tau] \tag{207}$$

$$\Psi''(\tau) = e^{-2\tau} [(4-\pi^2) \cos \pi\tau + 4\pi \sin \pi\tau] \tag{208}$$

$$\Psi'''(\tau) = e^{-2\tau} [(-8+6\pi^2) \cos \pi\tau + (\pi^2-12) \pi \sin \pi\tau] \tag{209}$$

Putting these values in Eq. (184) for σ results in

$$\sigma \approx \left.\frac{16\ \pi^2\ e^{-4\tau}}{8\ \pi^2\ e^{-4\tau}}\right|_{\tau>>1} = 2 \tag{210}$$

Similarly, after simplifying the trigonometric terms generated from substituting Eqs. (207)-(210) in Eq. (184) for β^2, β^2 becomes

$$\beta^2 \approx -\frac{-2(4-3\pi^2)\ e^{-2\tau}\ \cos\pi\tau + (\pi^2-12)\ \pi\ e^{-2\tau}\ \sin\pi\tau + \ldots}{-2\ e^{-2\tau}\ \cos\pi\tau - \pi\ e^{-2\tau}\ \sin\pi\tau}$$

$$\left.\frac{4(4-\pi^2)\ e^{-2\tau}\ \cos\pi\tau + 16\ \pi\ e^{-2\tau}\ \sin\pi\tau + \ldots}{3\ e^{-2\tau}\ \cos\pi\tau - 4\pi\ e^{-2\tau}\ \sin\pi\tau}\right|_{\tau>>1} \tag{211}$$

or

$$\beta^2 \approx \left.\frac{2\pi^2\ e^{-2\tau}(\cos\pi\tau + \pi\ \sin\pi\tau)}{2\ e^{-2\tau}(\cos\pi\tau + \pi\ \sin\pi\tau)}\right|_{\tau>>1} = \pi^2 \tag{212}$$

Consequently, the principal root of Eq. (212) gives

$$\beta \approx \pi \tag{213}$$

For $\Psi(\tau)$ prescribed in Eq. (199), the values $\sigma \approx 2$ and $\beta \approx \pi$ just derived have special intuitive appeal. When these values are used, and when a mean-square-error tolerance, Eq. (188), of $\delta \approx 0.9 \times 10^{-2}$ is selected, it is necessary to compute twelve expansion coefficients. Thus, with $N = 12$, $\sigma = 2$, and $\beta = \pi$, the a_k of Eq. (189) become

$$a_k = \sqrt{2\pi}\ \sum_{m=1}^{k} \lambda^*_{km}\ G_m = b_k \qquad k = 1,2,\ldots,N \tag{214}$$

where now

$$G_m = \int_0^\infty \Psi(\tau)\, e^{-m(\sigma+j\beta)\tau}\, d\tau = \int_0^\infty e^{-[m(\sigma+j\beta)+2]\tau} \cos \pi\tau\, d\tau \tag{215}$$

or

$$G_m = \frac{m(\sigma+j\beta) + 2}{[m(\sigma+j\beta)+2]^2 + \pi^2} \qquad m = 1,2,\ldots,N \tag{216}$$

Thus, the expansion coefficients, rounded to four significant digits,† are

$$\begin{array}{lll} a_1 = 0.3221 - j\,0.1132 & a_5 = -0.0316 + j\,0.0488 & a_9 = 0.0347 - j\,0.0048 \\ a_2 = -0.0967 + j\,0.0452 & a_6 = 0.0500 - j\,0.0014 & a_{10} = -0.0197 - j\,0.0249 \\ a_3 = 0.0798 + j\,0.0281 & a_7 = -0.0249 - j\,0.0361 & a_{11} = -0.0107 + j\,0.0270 \\ a_4 = -0.0242 - j\,0.0648 & a_8 = -0.0154 + j\,0.0358 & a_{12} = 0.0265 - j\,0.0037 \end{array} \tag{217}$$

with

$$\epsilon_{12} = 0.895 \times 10^{-2} < \delta \tag{218}$$

and

$$\epsilon_{11} = 0.934 \times 10^{-2} > \delta \tag{219}$$

The corresponding approximation $\hat{\Psi}(\tau)$ and the given autocorrelation function $\Psi(\tau)$ are compared in Figs. 4-7.

Finally, since the Wiener-Khintchine Theorem(37) relates the autocorrelation function and spectral density as Fourier transform pairs, the power density spectrum associated with the approximation

$$\hat{\Psi}_+(\tau) = \sum_{k=1}^{12} a_k X_k(\tau) \approx \Psi(\tau) \qquad \tau \geq 0 \tag{220}$$

---

† Sixteen significant digits were computed in all the calculations of a_k since the λ_{mn} may become large for certain values of σ, β, and m.

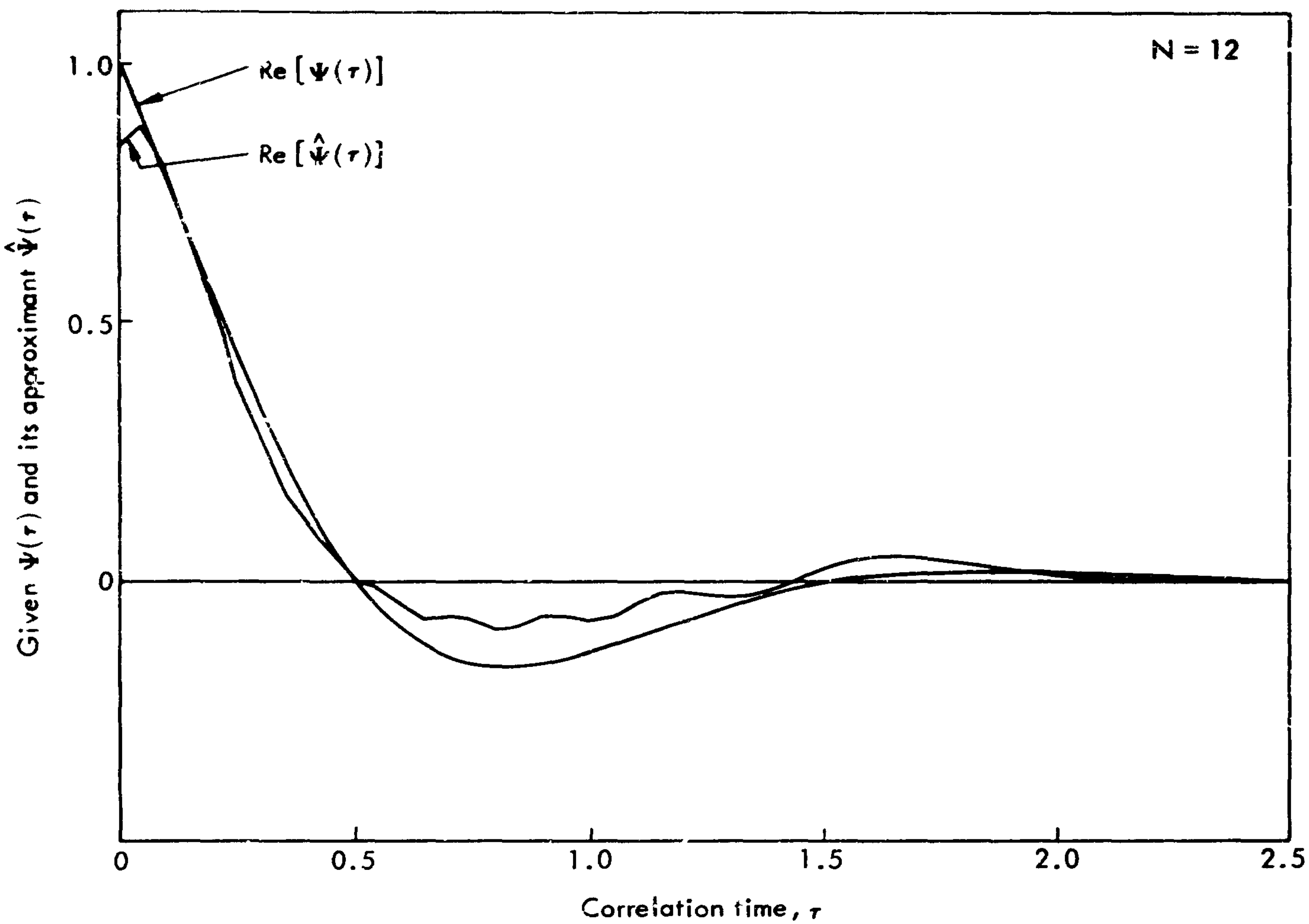

Fig. 4 — Graphs of the real parts of $\Psi(\tau)$ and $\hat{\Psi}(\tau)$ for N = 12

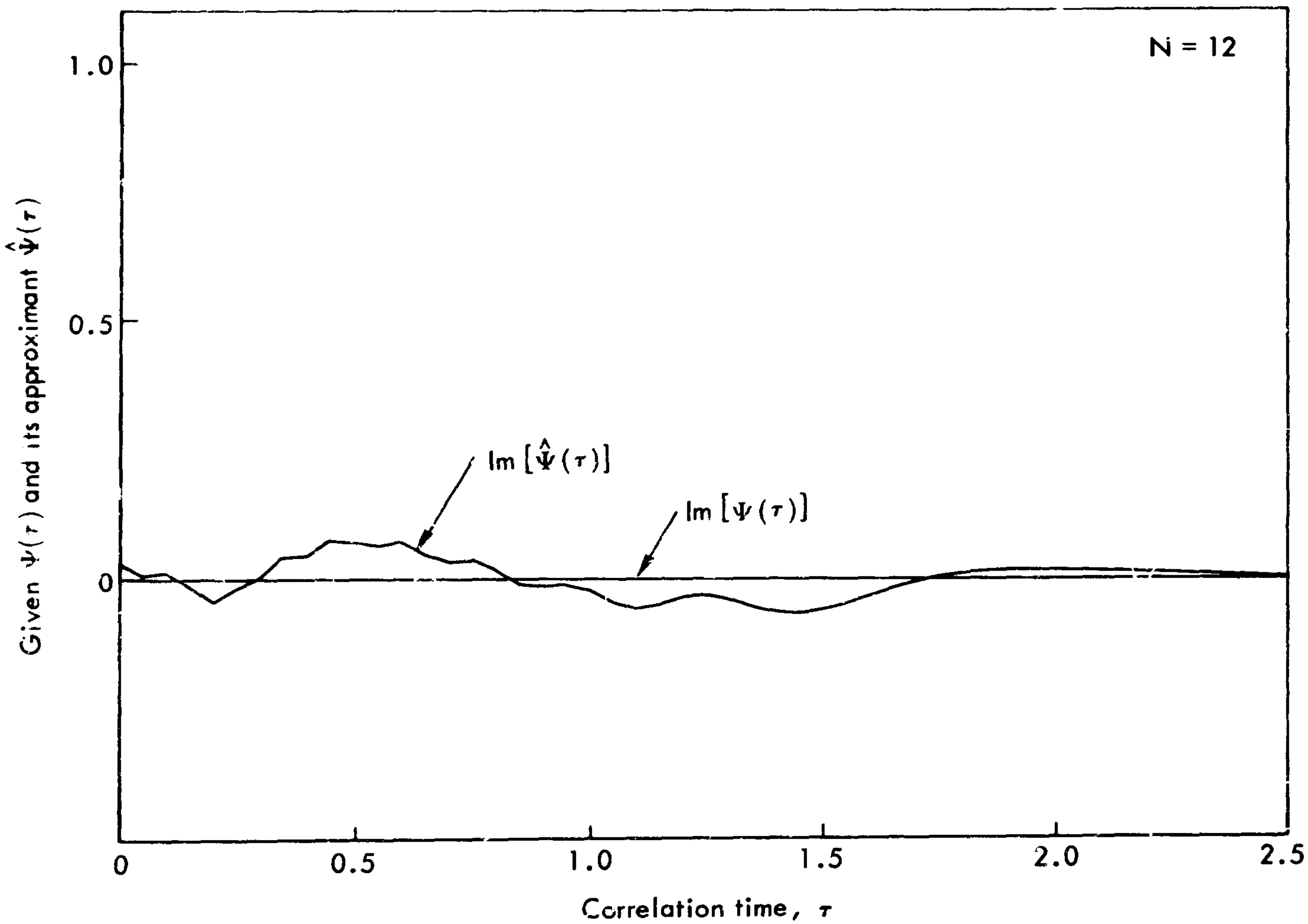

Fig.5-- Graphs of the imaginary parts of $\Psi(\tau)$ and $\hat{\Psi}(\tau)$ for N = 12

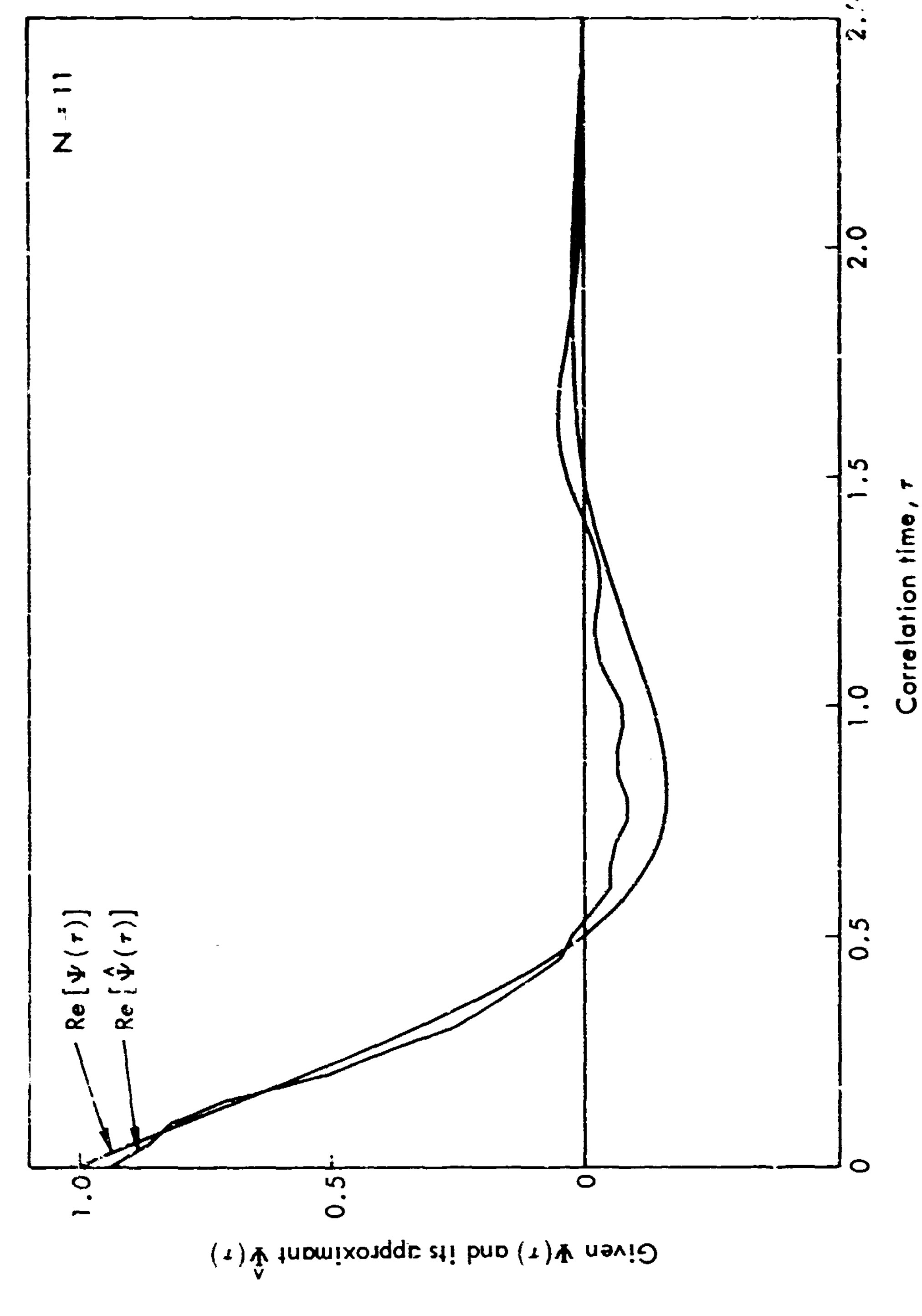

Fig.6 — Graphs of the real parts of $\Psi(\tau)$ and $\hat{\Psi}(\tau)$ for N = 11

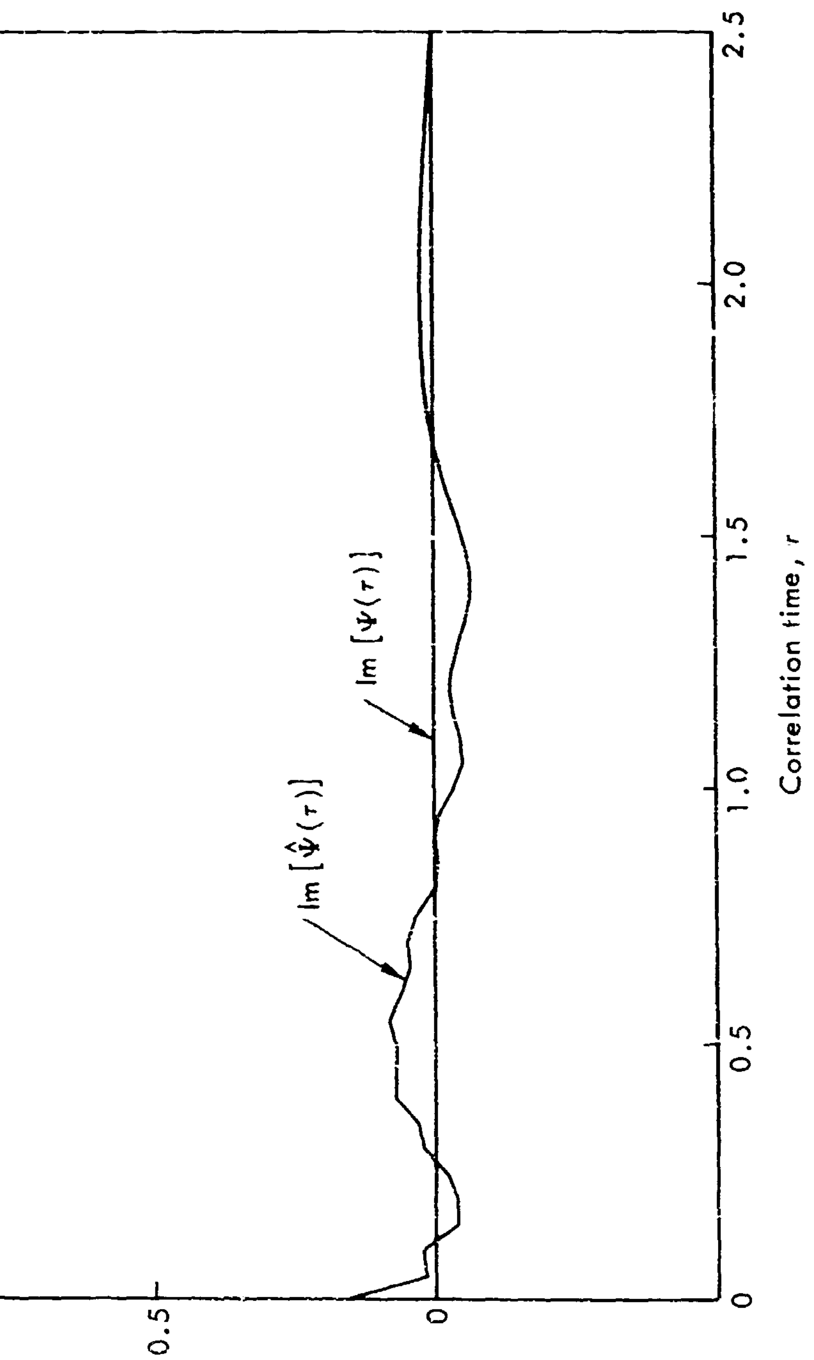

Fig. 7 — Graphs of the imaginary parts of $\Psi(\tau)$ and $\hat{\Psi}(\tau)$ for N = 11

$$\hat{\Psi}_-(\tau) = \sum_{k=1}^{12} a_k\, X_k(-\tau) \approx \Psi(\tau) \qquad \tau \le 0 \tag{221}$$

is

$$\hat{\Phi}(\omega) = \int_0^\infty \hat{\Psi}_+(\tau)\, e^{-j\omega\tau}\, d\tau + \int_{-\infty}^0 \hat{\Psi}_-(\tau)\, e^{-j\omega\tau}\, d\tau \tag{222}$$

$$\approx \int_{-\infty}^\infty \Psi(\tau)\, e^{-j\omega\tau}\, d\tau = \Phi(\omega)$$

Thus, by changing the variable in the second integral of Eq. (222) and by noting Eq. (221), Eq. (222) becomes

$$\Phi(\omega) \approx \sum_{k=1}^{12} a_k \left\{\int_0^\infty X_k(\tau)\, e^{-j\omega\tau}\, d\tau + \int_0^\infty X_k(\tau)\, e^{j\omega\tau}\, d\tau\right\} \tag{223}$$

Since $X_k(\tau)$ and $U_k(\omega)$ are related as Fourier transforms (to within a scale factor; see Eq. (23)), Eq. (223) can be written as

$$\Phi(\omega) \approx \sum_{k=1}^{12} \sqrt{2\pi}\, a_k\, [U_m(\omega) + U_m^*(\omega)] \tag{224}$$

or, in view of Eqs. (21) and (4), as

$$\Phi(\omega) \approx 2\sqrt{2\pi} \sum_{k=1}^{12} \sum_{n=1}^{k} a_k \left\{\frac{n\sigma\, \mathrm{Re}[\lambda_{kn}] + (\omega - n\beta)\, \mathrm{Im}[\lambda_{kn}]}{(n\sigma)^2 + (\omega - n\beta)^2}\right\} = \hat{\Phi}(\omega) \tag{225}$$

Thus, for any value of N, once the a_k are determined for the autocorrelation function $\Psi(\tau)$, the approximating spectral density can be written by inspection. In view of Eq. (145), moreover, the integral-square error between $\Phi(\omega)$ and $\hat{\Phi}(\omega)$ in Eq. (225) is simply $2\pi\epsilon_N$.

Appendix A

THE ORTHONORMAL BASIS COEFFICIENTS λ_{mn}

The rational function of σ and β given in Eq. (45) can be used to generate the coefficients λ_{mn}. As indicated in Section VII, however, there is considerable computational advantage in recursively evaluating λ_{mn} by Eq. (153). A program for doing this is presented in Appendix B.

In order to tabulate the algebraic λ_{mn} obtained by computer, the following definitions are convenient. From Eq. (45), λ_{mn} is expressed as the proper rational function

$$\lambda_{mn} = \begin{cases} \sqrt{\dfrac{\sigma}{\pi}} & n = m = 1 \\ \sqrt{\dfrac{m\sigma}{\pi}} \dfrac{\prod\limits_{r=1}^{m-1} [n(\sigma - j\beta) + r(\sigma + j\beta)]}{(\sigma - j\beta)^{m-1} \prod\limits_{r=1}^{m}{}' (r-n)} & \begin{array}{l} m = 2,3,\ldots \\ 1 \le n \le m \end{array} \\ 0 & \begin{array}{l} m = 1,2,\ldots \\ n > m \end{array} \end{cases} \tag{226}$$

If η and Λ_{mn} are defined as

$$\eta = j\beta \tag{227}$$

and

$$\Lambda_{mn} = \frac{\prod\limits_{r=1}^{m-1} [n(\sigma - j\beta) + r(\sigma + j\beta)]}{\prod\limits_{r=1}^{m}{}' (r-n)} = \frac{\prod\limits_{r=1}^{m-1} [n(\sigma - \eta) + r(\sigma + \eta)]}{\prod\limits_{r=1}^{m}{}' (r-n)} = (\sigma - \eta)^{m-1} \alpha_{mn} \tag{228}$$

where α_{mn} is given by Eq. (68), then λ_{mn} can be reexpressed in terms of the polynomial Λ_{mn} in σ and η as

$$\lambda_{mn} = \begin{cases} \sqrt{\dfrac{\sigma}{\pi}} & n = m = 1 \\ \dfrac{\sqrt{\dfrac{m\sigma}{\pi}}}{(\sigma-\eta)^{m-1}} \Lambda_{mn} & \begin{matrix} m = 2,3,\ldots \\ 1 \le n \le m \end{matrix} \\ 0 & \begin{matrix} m = 1,2,\ldots \\ n > m \end{matrix} \end{cases} \qquad (229)$$

The quantities Λ_{mn} defined by Eq. (228) are shown in Table 4 for the range $m = 1,2,\ldots,10$ and $n = 1,2,\ldots,m$. For each value of m, the check-sum relation Eq. (67) holds, thereby verifying the exactness of the rational λ_{mn} derived from the polynomials Λ_{mn}.

The Λ_{mn} of Table 4 have been generated by encoding the recursive relation, Eq. (153), in ALTRAN, a symbolic manipulation language, and executing the program on a digital computer.[38] The ALTRAN compiler, which is required for execution of the program listed in Appendix B, produces MAP output consisting of transfers to ALPAK, a group of subroutines for computing and simplifying polynomials and certain rational functions.

Since ALTRAN is not widely available and since it requires large computer storage for evaluating Λ_{mn} for m as high as ten, another program is given in Appendix C for computing the Λ_{mn}. The Fortran IV routine listed in Appendix C computes the integer coefficients of the variables σ and β in the polynomials Λ_{mn}. The computational basis

for the program is discussed in Appendix D where a series representation for Λ_{mn} is derived to replace the product form given by Eq. (228). Finally, in Appendix E a Fortran IV program is presented which enables arbitrary functions $g(t) \subset L^2(0,\infty)$ to be expanded in terms of the orthonormal elements $X_m(t)$.

Table 4

THE ASSOCIATED ORTHONORMAL BASIS COEFFICIENTS Λ_{mn}

$$\Lambda_{11} = 1$$

$$\Lambda_{21} = 2\sigma$$

$$\Lambda_{22} = -3\sigma + \eta$$

$$\Lambda_{31} = 3\sigma^2 + \sigma\eta$$

$$\Lambda_{32} = -12\sigma^2 + 4\sigma\eta$$

$$\Lambda_{33} = 10\sigma^2 - 7\sigma\eta + \eta^2$$

$$\Lambda_{41} = (12\sigma^3 + 10\sigma^2\eta + 2\sigma\eta^2)/3$$

$$\Lambda_{42} = -30\sigma^3 + 4\sigma^2\eta + 2\sigma\eta^2$$

$$\Lambda_{43} = 60\sigma^3 - 42\sigma^2\eta + 6\sigma\eta^2$$

$$\Lambda_{44} = (-105\sigma^3 + 113\sigma^2\eta - 35\sigma\eta^2 + 3\eta^3)/3$$

$$\Lambda_{51} = (30\sigma^4 + 43\sigma^3\eta + 20\sigma^2\eta^2 + 3\sigma\eta^3)/6$$

$$\Lambda_{52} = (-180\sigma^4 - 36\sigma^3\eta + 20\sigma^2\eta^2 + 4\sigma\eta^3)/3$$

$$\Lambda_{53} = 210\sigma^4 - 117\sigma^3\eta + 3\sigma\eta^3$$

$$\Lambda_{54} = (-840\sigma^4 + 904\sigma^3\eta - 280\sigma^2\eta^2 + 24\sigma\eta^3)/3$$

$$\Lambda_{55} = (756\sigma^4 - 1101\sigma^3\eta + 536\sigma^2\eta^2 - 101\sigma\eta^3 + 6\eta^4)/6$$

$$\Lambda_{61} = (90\sigma^5 + 189\sigma^4\eta + 146\sigma^3\eta^2 + 49\sigma^2\eta^3 + 6\sigma\eta^4)/15$$

$$\Lambda_{62} = (-315\sigma^5 - 198\sigma^4\eta + 8\sigma^3\eta^2 + 22\sigma^2\eta^3 + 3\sigma\eta^4)/3$$

$$\Lambda_{63} = 560\sigma^5 - 172\sigma^4\eta - 78\sigma^3\eta^2 + 8\sigma^2\eta^3 + 2\sigma\eta^4$$

Table 4--continued

$$\Lambda_{64} = (-3780\sigma^5 + 3648\sigma^4\eta - 808\sigma^3\eta^2 - 32\sigma^2\eta^3 + 12\sigma\eta^4)/3$$

$$\Lambda_{65} = (3780\sigma^5 - 5505\sigma^4\eta + 2680\sigma^3\eta^2 - 505\sigma^2\eta^3 + 30\sigma\eta^4)/3$$

$$\Lambda_{66} = (-2310\sigma^5 + 4247\sigma^4\eta - 2842\sigma^3\eta^2 + 852\sigma^2\eta^3 - 112\sigma\eta^4 + 5\eta^5)/5$$

$$\Lambda_{71} = (630\sigma^6 + 1773\sigma^5\eta + 1967\sigma^4\eta^2 + 1073\sigma^3\eta^3 + 287\sigma^2\eta^4 + 30\sigma\eta^5)/90$$

$$\Lambda_{72} = (-2520\sigma^6 - 2844\sigma^5\eta - 728\sigma^4\eta^2 + 208\sigma^3\eta^3 + 112\sigma^2\eta^4 + 12\sigma\eta^5)/15$$

$$\Lambda_{73} = (2520\sigma^6 + 66\sigma^5\eta - 609\sigma^4\eta^2 - 81\sigma^3\eta^3 + 21\sigma^2\eta^4 + 3\sigma\eta^5)/2$$

$$\Lambda_{74} = (-37800\sigma^6 + 28920\sigma^5\eta - 784\sigma^4\eta^2 - 1936\sigma^3\eta^3 + 56\sigma^2\eta^4 + 24\sigma\eta^5)/9$$

$$\Lambda_{75} = (41580\sigma^6 - 56775\sigma^5\eta + 23975\sigma^4\eta^2 - 2875\sigma^3\eta^3 - 175\sigma^2\eta^4 + 30\sigma\eta^5)/6$$

$$\Lambda_{76} = (-27720\sigma^6 + 50964\sigma^5\eta - 34104\sigma^4\eta^2 + 10224\sigma^3\eta^3 - 1344\sigma^2\eta^4 + 60\sigma\eta^5)/5$$

$$\Lambda_{77} = (154440\sigma^6 - 343146\sigma^5\eta + 293243\sigma^4\eta^2 - 122023\sigma^3\eta^3 + 25703\sigma^2\eta^4 - 2547\sigma\eta^5 + 90\eta^6)/90$$

$$\Lambda_{81} = (2520\sigma^7 + 8982\sigma^6\eta + 13187\sigma^5\eta^2 + 10193\sigma^4\eta^3 + 4367\sigma^3\eta^4 + 981\sigma^2\eta^5 + 90\sigma\eta^6)/315$$

$$\Lambda_{82} = (-11340\sigma^7 - 19098\sigma^6\eta - 10386\sigma^5\eta^2 - 884\sigma^4\eta^3 + 1024\sigma^3\eta^4 + 334\sigma^2\eta^5 + 30\sigma\eta^6)/45$$

$$\Lambda_{83} = (12600\sigma^7 + 5370\sigma^6\eta - 2913\sigma^5\eta^2 - 1623\sigma^4\eta^3 - 57\sigma^3\eta^4 + 57\sigma^2\eta^5 + 6\sigma\eta^6)/5$$

$$\Lambda_{84} = (-103950\sigma^7 + 51180\sigma^6\eta + 19534\sigma^5\eta^2 - 5912\sigma^4\eta^3 - 1298\sigma^3\eta^4 + 108\sigma^2\eta^5 + 18\sigma\eta^6)/9$$

Table 4--continued

$$\Lambda_{85} = (249480\sigma^7 - 299070\sigma^6\eta + 87075\sigma^5\eta^2 + 6725\sigma^4\eta^3 - 3925\sigma^3\eta^4 + 5\sigma^2\eta^5 + 30\sigma\eta^6)/9$$

$$\Lambda_{86} = (-180180\sigma^7 + 317406\sigma^6\eta - 196194\sigma^5\eta^2 + 49404\sigma^4\eta^3 - 3624\sigma^3\eta^4 - 282\sigma^2\eta^5 + 30\sigma\eta^6)/5$$

$$\Lambda_{87} = (1081080\sigma^7 - 2402022\sigma^6\eta + 2052701\sigma^5\eta^2 - 854161\sigma^4\eta^3 + 179921\sigma^3\eta^4 - 17829\sigma^2\eta^5 + 630\sigma\eta^6)/45$$

$$\Lambda_{88} = (-2027025\sigma^7 + 5282325\sigma^6\eta - 5503581\sigma^5\eta^2 + 2947489\sigma^4\eta^3 - 867299\sigma^3\eta^4 + 138319\sigma^2\eta^5 - 10863\sigma\eta^6 + 315\eta^7)/315$$

$$\Lambda_{91} = (22680\sigma^8 + 98478\sigma^7\eta + 181557\sigma^6\eta^2 + 184046\sigma^5\eta^3 + 110654\sigma^4\eta^4 + 39398\sigma^3\eta^5 + 7677\sigma^2\eta^6 + 630\sigma\eta^7)/2520$$

$$\Lambda_{92} = (-113400\sigma^8 - 259020\sigma^7\eta - 218448\sigma^6\eta^2 - 71156\sigma^5\eta^3 + 4936\sigma^4\eta^4 + 9484\sigma^3\eta^5 + 2304\sigma^2\eta^6 + 180\sigma\eta^7)/315$$

$$\Lambda_{93} = (46200\sigma^8 + 40690\sigma^7\eta - 1731\sigma^6\eta^2 - 10806\sigma^5\eta^3 - 2914\sigma^4\eta^4 + 114\sigma^3\eta^5 + 117\sigma^2\eta^6 + 10\sigma\eta^7)/10$$

$$\Lambda_{94} = (-1247400\sigma^8 + 198360\sigma^7\eta + 439128\sigma^6\eta^2 + 7192\sigma^5\eta^3 - 39224\sigma^4\eta^4 - 3896\sigma^3\eta^5 + 648\sigma^2\eta^6 + 72\sigma\eta^7)/45$$

$$\Lambda_{95} = (3243240\sigma^8 - 3139470\sigma^7\eta + 234765\sigma^6\eta^2 + 348650\sigma^5\eta^3 - 30850\sigma^4\eta^4 - 11710\sigma^3\eta^5 + 405\sigma^2\eta^6 + 90\sigma\eta^7)/36$$

$$\Lambda_{96} = (-840840\sigma^8 + 1361108\sigma^7\eta - 703968\sigma^6\eta^2 + 99756\sigma^5\eta^3 + 16024\sigma^4\eta^4 - 3732\sigma^3\eta^5 - 48\sigma^2\eta^6 + 20\sigma\eta^7)/5$$

$$\Lambda_{97} = (16216200\sigma^8 - 34949250\sigma^7\eta + 2\text{[illegible]}388493\sigma^6\eta^2 - 10759714\sigma^5\eta^3 + 1844654\sigma^4\eta^4 - 87514\sigma^3\eta^5 - 8379\sigma^2\eta^6 + 630\sigma\eta^7)/90$$

Table 4--continued

$$\Lambda_{98} = (-32432400\sigma^8 + 84517200\sigma^7\eta - 88057296\sigma^6\eta^2 + 47159824\sigma^5\eta^3 - 13876784\sigma^4\eta^4 + 2213104\sigma^3\eta^5 - 173808\sigma^2\eta^6 + 5040\sigma\eta^7)/315$$

$$\Lambda_{99} = (6806800\sigma^8 - 20355850\sigma^7\eta + 25047613\sigma^6\eta^2 - 16458298\sigma^5\eta^3 + 6267166\sigma^4\eta^4 - 1402306\sigma^3\eta^5 + 177733\sigma^2\eta^6 - 11458\sigma\eta^7 + 280\eta^8)/280$$

$$\Lambda_{10\ 1} = (11340\sigma^9 + 583110\sigma^8\eta + 1301697\sigma^7\eta^2 + 1646458\sigma^6\eta^3 + 1289454\sigma^5\eta^4 + 639606\sigma^4\eta^5 + 195977\sigma^3\eta^6 + 33858\sigma^2\eta^7 + 2520\sigma\eta^8)/11340$$

$$\Lambda_{10\ 2} = (-311850\sigma^9 - 910755\sigma^8\eta - 1054017\sigma^7\eta^2 - 577963\sigma^6\eta^3 - 110949\sigma^5\eta^4 + 34719\sigma^4\eta^5 + 22933\sigma^3\eta^6 + 4527\sigma^2\eta^7 + 315\sigma\eta^8)/630$$

$$\Lambda_{10\ 3} = (277200\sigma^9 + 382740\sigma^8\eta + 111684\sigma^7\eta^2 - 70029\sigma^6\eta^3 - 49902\sigma^5\eta^4 - 8058\sigma^4\eta^5 + 1044\sigma^3\eta^6 + 411\sigma^2\eta^7 + 30\sigma\eta^8)/35$$

$$\Lambda_{10\ 4} = (-8108100\sigma^9 - 1829160\sigma^8\eta + 3350232\sigma^7\eta^2 + 1144568\sigma^6\eta^3 - 236976\sigma^5\eta^4 - 123384\sigma^4\eta^5 - 5528\sigma^3\eta^6 + 2088\sigma^2\eta^7 + 180\sigma\eta^8)/135$$

$$\Lambda_{10\ 5} = (4540536\sigma^9 - 3097962\sigma^8\eta - 927117\sigma^7\eta^2 + 582016\sigma^6\eta^3 + 96270\sigma^5\eta^4 - 28734\sigma^4\eta^5 - 4117\sigma^3\eta^6 + 288\sigma^2\eta^7 + 36\sigma\eta^8)/18$$

$$\Lambda_{10\ 6} = (-3153150\sigma^9 + 4473525\sigma^8\eta - 1619049\sigma^7\eta^2 - 153891\sigma^6\eta^3 + 134907\sigma^5\eta^4 - 1977\sigma^4\eta^5 - 2979\sigma^3\eta^6 + 39\sigma^2\eta^7 + 15\sigma\eta^8)/5$$

$$\Lambda_{10\ 7} = (129729600\sigma^9 - 263377800\sigma^8\eta + 192158694\sigma^7\eta^2 - 57689219\sigma^6\eta^3 + 3997518\sigma^5\eta^4 + 1144542\sigma^4\eta^5 - 154546\sigma^3\eta^6 - 3339\sigma^2\eta^7 + 630\sigma\eta^8)/135$$

$$\Lambda_{10\ 8} = (2756754000\sigma^9 + 702180000\sigma^8\eta - 706228416\sigma^7\eta^2 + 356829856\sigma^6\eta^3 - 94372752\sigma^5\eta^4 + 11872992\sigma^4\eta^5 - 370816\sigma^3\eta^6 - 44064\sigma^2\eta^7 + 2520\sigma\eta^8)/315$$

Table 4--continued

$$\Lambda_{10\ 9} = (612612000\sigma^9 - 183202650\sigma^8\eta + 225428517\sigma^7\eta^2 - 148124682\sigma^6\eta^3$$
$$+ 564044494\sigma^5\eta^4 - 126207540\sigma^4\eta^5 + 1599597\sigma^3\eta^6 - 103122\sigma^2\eta^7$$
$$+ 2520\sigma\eta^8)/140$$

$$\Lambda_{10\ 10} = (-1047566520\sigma^9 + 3535985430\sigma^8\eta - 5021073090\sigma^7\eta^2 + 3916728650\sigma^6\eta^3$$
$$- 1838402610\sigma^5\eta^4 + 534437430\sigma^4\eta^5 - 9532127\sigma^3\eta^6 + 993411\sigma^2\eta^7$$
$$- 53955\sigma\eta^8 + 1134\eta^9)/1134$$

Appendix B

AN ALTRAN PROGRAM FOR GENERATING Λ_{mn}

The ALTRAN program listed in this appendix is used for computing the polynomials Λ_{mn} of Eq. (228). A recurrence relation similar to Eq. (153) is the basis for the routine.

In order to execute this program, an ALTRAN compiler is required to produce MAP and, thereby, the final object code.[†] The accompanying routine and its associated control statements are appropriate for execution on The RAND Corporation's IBM 7044 computer. The prologue to the listings describes the program's parameters, usage, and limitations.

[†] The ALTRAN compiler consists of transfers to ALPAK, a group of subroutines for operating on certain polynomials and rational functions. The details of ALTRAN and ALPAK, including the format for representing polynomials, are discussed in Ref. 38.

```
C
C
C      THE FOLLOWING LISTING IS AN ALTRAN PROGRAM FOR COMPUTING
C      THE POLYNOMIALS LAMBDA(M,N) IN THE VARIABLES SIGMA AND BETA.
C      THE POLYNOMIALS ARE EXPRESSED AS CONSTANTS TIMES (SIGMA**(M-I))X
C      (BETA**I) FOR I=0,1,...,M AND FOR M=1,2,... AND FOR N=1,2,...,M
C      THE LAMBDA(M,N) ARE ZERO FOR N GREATER THAN M.
C      THE FOLLOWING PROGRAM CALCULATES LAMBDA(M,N) FOR M=1,2,...,10.
C      THE CONTROL CARDS LISTED BELOW INDICATE THE APPROPRIATE DECK
C      SET UP FOR EXECUTION ON THE IBM 7044. BINARY DECKS ARE NOT
C      FULLY LISTED. THE LAMBDA(M,N) ARE MADE AVAILABLE AS BOTH
C      PRINTED AND PUNCHED CARD OUTPUT.
C
C
$CLOSE          S.SU07,REWIND
$IBJOB          MAP,FILES
$FILE           'S.FBIA',NONE,*,BLOCK=10
$FILE           'S.FBOA',NONE,*,BLOCK=10
$IEDIT          U07,SRCH
$IBLDR TMG
$IBLDR TMGIO
$IBLDR TMGDFN
$IBLDR ALTRAN
$ENTRY           TMG
      STORAGE 13000
      LAYOUT (L) SIGMA 18, GAMMA 18
      ALGEBRAIC (L) LAMBDA(10,10)
      INTEGER M,N,R,I
      LAMBDA(1,1)=1
      LAMBDA(2,1)=2*SIGMA
      LAMBDA(2,2)=-3*SIGMA+GAMMA
      PRINT LAMBDA(1,1),LAMBDA(2,1),LAMBDA(2,2)
      PUNCH LAMBDA(1,1),LAMBDA(2,1),LAMBDA(2,2)
      DO 20 M=3,10
      I=M-1
      DO 15 N=1,I
      LAMBDA(M,N)=LAMBDA(M-1,N)*(N*(SIGMA-GAMMA)+I*(SIGMA+GAMMA))/(M-N)
      PRINT M,N,LAMBDA(M,N)
      PUNCH     LAMBDA(M,N)
   15 CONTINUE
      LAMBDA(M,M)=1
      DO 16 R=1,I
      LAMBDA(M,M)=LAMBDA(M,M)*(M*(SIGMA-GAMMA)+R*(SIGMA+GAMMA))/(R-M)
   16 CONTINUE
      PRINT   LAMBDA(M,M)
      PUNCH   LAMBDA(M,M)
   20 CONTINUE
      STOP
      END
      FINISH

$IBSYS
$CLOSE          S.SU06,REWIND
$IBJOB          MAP,IOOP1
$IEDIT          U06,SRCH
```

```
$IBMAP ALTRAN
$IEDIT          U07,SRCH
$IBLDR ALFSRT
$IBLDR READF
$IBLDR READD
$IBLDR READI
$IBLDR OUT
$IBLDR PUNCHP
$IBLDR ALF
$IBLDR ALP
$IEDIT          IN
$IBLDR IORED    10/28/66
$CDICT IORED
C      BINARY CARDS DELETED
$TEXT  IORED
C      BINARY CARDS DELETED
$DKEND IORED
$IBLDR POSTXX   10/28/66
$CDICT POSTXX
C      BINARY CARDS DELETED
$TEXT  POSTXX
C      BINARY CARDS DELETED
$DKEND POSTXX
$ENTRY          ALT
$IBSYS
$CLOSE          S.SU07,REMOVE
```

Appendix C

A FORTRAN IV PROGRAM FOR GENERATING THE COEFFICIENTS IN Λ_{mn}

The FORTRAN IV program listed in this appendix is used to compute the constants in the polynomials Λ_{mn} of Eq. (228). The routine is based on Eqs. (237), (240), and (241) of Appendix D.

The following program is compatible with the IBM 7044, 7094, and 360 series FORTRAN IV compilers. The program obviates the large storage needed in the ALTRAN routine discussed in Appendix B. The routine also allows cross-checking with the results of ALTRAN.

The prologue to the listing describes the parameters, usage, and limitations of the program, as well as the format of the printed results. All of the computation is performed in double precision.

```
$IBFTC COEFF
C
C
C     THIS ROUTINE PRODUCES THE MATRIX OF LAMBDA(M,N) COEFFICIENTS.
C     M RANGES FROM 1 TO 15 AND N RANGES FROM 1 TO M.
C     THE COEFFICIENTS ARE ZERO FOR ALL N GREATER THAN M.
C     THE COEFFICIENTS OF THE VARIABLES SIGMA**I X BETA**J ARE GIVEN
C     FOR EACH POLYNOMIAL LAMBDA(M,N). THE OUTPUT FORMAT IS AS FOLLOWS
C     M=  ,N=  , POWER OF SIGMA=  , POWER OF BETA=  ,  COEFFICIENT=  .
C     ALL COMPUTATION IS IN DOUBLE PRECISION. FOR M GREATER THAN 15
C     HIGHER PRECISION IS REQUIRED, THE INDEXING MUST BE INCREASED
C     AND THE DIMENSION STATEMENTS MUST BE ADJUSTED.
C
C
      DOUBLE PRECISION S(16,16),F(16),AM,AN,A,ANS,B,C
      INTEGER R,P
      COMPLEX  Q
      LOGICAL TEST
      WRITE(6,6)
    6 FORMAT(1H1////10X,1HM,3X,     1HN,3X,14HPOWER OF SIGMA,3X,
     .13HPOWER OF BETA,3X,30HCOEFFICIENT OF SIGMA-BETA TERM  ////)
      DO 1 M=1,15
      S(M,M)=1.D0
      AM=DBLE(FLOAT(M))
      S(M+1,1)=-AM*S(M,1)
      M1=M+1
      DO 1 K=M1,15
      S(M,K)=0.D0
    1 CONTINUE
      DO 4 M=2,15
      AM=DBLE(FLOAT(M))
      DO 4 K=2,15
    4 S(M+1,K)=S(M,K-1)-AM*S(M,K)
      F(1)=1.D0
      F(2)=F(1)
      DO 2 I=2,14
    2 F(I+1)=F(I)*DBLE(FLOAT(I))
      DO 100 M=1,10
      AM=DBLE(FLOAT(M))
      M2=M-1
      DO 100 N=1,M
      AN=DBLE(FLOAT(N))
      MN=M-N+1
      A=((-1.D0)**(M+N))/(F(N+1)*F(MN))
      DO 100 II=1,M
      ISIGMA=IABS(M-II)
      IF(II.EQ.M.AND.N.NE.M) GO TO 100
      IBETA=IABS(M2-ISIGMA)
      ANS=0.D0
      DO 3 K=1,M
      MK=M-K +1
      B=(-AN)**K*S(M,K)*F(MK)*F(K)
      DO 3 R=1,MK
      MKR=M-K-R+2
      DO 3 P=1,K
```

```
      C=B*(-1.D0)**(P-1)
      KP=K-P+1
      IF((M-R-P+1).EQ.ISIGMA) ANS=ANS+C/(F(R)*F(P)*F(MKR)*F(KP))
    3 CONTINUE
      ANS=-ANS*A
      Q=CMPLX(0.,1.)**IBETA
      IQ1=REAL(Q)
      IQ2=AIMAG(Q)
      TEST=IQ2.EQ.0.AND.IQ1.LT.0
      IF(TEST) ANS=-ANS
      IF(TEST) WRITE(6,7) M,N,ISIGMA,IBETA,ANS
    7 FORMAT(9X,I2,2X,I2,9X,I2,15X,I2,11X,D24.16)
      IF(TEST) GO TO 100
      TEST=IQ1.EQ.0.AND.IQ2.LT.0
      IF(TEST) ANS=-ANS
      IF(TEST) WRITE(6,5) M,N,ISIGMA,IBETA,ANS
    5 FORMAT(9X,I2,2X,I2,9X,I2,15X,I2,11X,D24.16,2X,1HI)
      IF(TEST) GO TO 100
      IF(IQ1.EQ.0.AND.IQ2.GE.0) WRITE(6,5) M,N,ISIGMA,IBETA,ANS
      IF(IQ2.EQ.0.AND.IQ1.GE.0) WRITE(6,7) M,N,ISIGMA,IBETA,ANS
  100 CONTINUE
      CALL EXIT
      END
```

Appendix D

A SERIES REPRESENTATION FOR Λ_{mn}

For reasons discussed in Appendix A, a series representation for Λ_{mn} is a useful supplement to the product form given in Eq. (228).

Starting with Eqs. (68) and (228) for α_{mn} and Λ_{mn}, one can write

$$\alpha_{mn} = \sqrt{\frac{\pi}{m\sigma}}\ \lambda_{mn} \tag{230}$$

and

$$\Lambda_{mn} = (\sigma - j\beta)^{m-1}\ \alpha_{mn} \tag{231}$$

From Eq. (78) it follows that

$$\alpha_{mn} = \frac{(-1)^{m+1}}{m!} \sum_{k=0}^{m} (-1)^{k+n} \binom{m}{n} S_m^{(k)}\ n^k\ z^{m-k} \tag{232}$$

where z is defined by Eq. (71), and $S_m^{(k)}$ are the Stirling numbers of the first kind given by Eq. (73).

Thus

$$\Lambda_{mn} = \frac{(-1)^{m+n+1}}{(m-n)!\ n!} \sum_{k=0}^{m} (-1)^k\ S_m^{(k)}\ n^k\ (\sigma + j\beta)^{m-k}\ (\sigma - j\beta)^{k-1} \tag{233}$$

The binominal theorem[37] indicates that

$$(\sigma + j\beta)^{m-k} = \sum_{r=0}^{m-k} \sigma^{m-k-r}\ (j\beta)^r \binom{m-k}{r} \tag{234}$$

and

$$(\sigma - j\beta)^{k-1} \sum_{p=0}^{k-1} \sigma^{k-1-p}\ (-j\beta)^p \binom{k-1}{p} \tag{235}$$

Substitution of Eqs. (234) and (235) in Eq. (233) therefore results in

$$\Lambda_{mn} = \frac{(-1)^{m+n+1}}{n!(m-n)!} \sum_{k=0}^{m} \sum_{r=0}^{m-k} \sum_{p=0}^{k-1} \left\{ \frac{(-1)^{k+p} S_m^{(k)} n^k (m-k)!(k-1)!}{r!p!(m-k-r)!(k-p-1)!} \right\} \sigma^{m-r-p-1} (j\beta)^{r+p} \tag{236}$$

Since $S_m^{(0)} = 0$, the series representation for Λ_{mn} simplifies to

$$\Lambda_{mn} = \frac{(-1)^{m+n+1}}{n!(m-n)!} \sum_{k=1}^{m} (-1)^k S_m^{(k)} n^k (m-k)!(k-1)! \sum_{r=1}^{m-k+1} \sum_{p=1}^{k}$$

$$\left\{ \frac{(-1)^p (j)^{r+p}}{(r-1)!(p-1)!(k-p)!(m-k-r+1)!} \right\} \sigma^{m-r-p+1} \beta^{r+p-2} \tag{237}$$

In order to obtain the coefficients c_i in the representation

$$\Lambda_{mn} = \sum_{i=0}^{m-1} c_i \sigma^{m-1-i} \beta^i \tag{238}$$

which follows from Eq. (237), one notes that r and p must be selected according to

$$\left\{ \begin{array}{ll} \sigma^{m-1-i} = \sigma^{m-r-p+1} & r = 1,2,..,m-k+1 \\ & p = 1,2,\ldots,k \\ \beta^i = \beta^{r+p-2} & k = 1,2,\ldots,m \end{array} \right. \tag{239}$$

Thus, for any value of i and r

$$m-1-i = m-r-p+1 \qquad i = 0,1,\ldots,m-1;\ k = 1,2,\ldots,m;\ r = 1,2,\ldots,m-k+1 \tag{240}$$

so that p is constrained to the value(s)

$$p = i-r+2 \qquad \text{with } p \in \{1,2,\ldots,k\} \tag{241}$$

Consequently, c_i is the aggregate of all terms of Eq. (237) obtained by allowing k to range over the integers from 1 to m; r, from 1 to m-k+1; and p = i-r+2, with p contained in the set k = 1,2,...,m. Such a summation is easily programmed for digital computation; a FORTRAN IV routine for finding the c_i is provided in Appendix C.

Appendix E

A FORTRAN IV PROGRAM FOR COMPUTING EXPONENTIAL APPROXIMATIONS $\hat{g}(t)$

The accompanying program is designed to calculate the exponential approximation $\hat{g}(t)$ to a prescribed function $g(t) \subset L^2(0,\infty)$. The algorithm is based on the recurrence relations Eqs. (157) and (158) for the m^{th} orthonormal basis function $X_m(t)$, on recurrence relation Eq. (153) for λ_{mn}, and on Eqs. (132) or (162) for the moments of g(t). These moments are approximated by a 64-point Gaussian quadrature scheme and are used to form the Fourier expansion coefficients a_m according to Eq. (131). The approximant $\hat{g}(t)$ is finally obtained from Eq. (133).

The numerical examples discussed in Section IX have been solved with the use of program APRX. The values of σ and β obtained from Eqs. (175), or from Eqs. (184) and (185), are input parameters to the program as described in the prologue of APRX. All other program options, definitions, and limitations are also clarified in the listings.

The real and imaginary parts of g(t) must be supplied as the double-precision function subroutines named GR(T) and GI(T), respectively. Once APRX is entered, GR, GI, and all the supporting routines are automatically invoked to produce a 51-point tabular and graphical display of $\hat{g}(t)$. Card output of $\hat{g}(t)$ for subsequent processing is also available.

```
$IBFTC APRX
C
C
C      THIS ROUTINE IS THE MAIN PROGRAM FOR OBTAINING AN EXPANSION OF
C      A GIVEN FUNCTION G(T) IN TERMS OF THE ORTHONORMAL SET X(M,T).
C      THE REAL PART OF G(T) MUST BE SUPPLIED AS A DOUBLE PRECISION
C      FUNCTION SUBROUTINE CALLED GR(T). AN EXAMPLE IS GIVEN AT THE
C      END OF THE LISTINGS. THE IMAGINARY PART OF G(T) MUST BE SUPPLIED
C      AS A DOUBLE PRECISON FUNCTION SUBROUTINE CALLED GI(T). AN EXAMPLE
C      IS GIVEN AT THE END OF THE LISTINGS. THE REPRESENTATION IS BEST
C      IN AN INTEGRAL SQUARE ERROR SENSE OVER THE INTERVAL ZERO TO
C      INFINITY. IN ORDER TO FACILITATE THE INTEGRATION AND PLOTTING
C      THE FOLLOWING PARAMETERS MUST BE READ IN FROM A DATA CARD
C      ACCORDING TO THE FORMAT NUMBER 8 GIVEN BELOW.
C      MC=THE NUMBER OF TERMS DESIRED IN THE EXPANSION.
C      SIG=THE VALUE CHOSEN FOR SIGMA.
C      BET=THE VALUE SELECTED FOR BETA.
C      AA=THE SMALLEST VALUE OF T (GREATER THAN OR EQUAL TO ZERO) BEFORE
C      WHICH THE PRESCRIBED FUNCTION G(T) IS ESSENTIALLY ZERO.
C      BB=THE LARGEST VALUE OF T (GREATER THAN ZERO) AFTER WHICH G(T)
C      IS ESSENTIALLY ZERO. AA AND BB ARE THE LIMITS USED IN COMPUTING
C      THE FOURIER COEFFICIENTS FOR THE EXPANSION ON ZERO TO INFINITY.
C      GRMAX=MAXIMUM VALUE OF THE REAL PART OF G(T) TO BE IN THE GRAPH.
C      GRMIN=MINIMUM VALUE OF THE REAL PART OF G(T) TO BE IN THE PLOT.
C      THE SAME ORDINATE SCALE WILL BE USED IN PLOTTING THE IMAGINARY PART.
C      TMAX=MAXIMUM ABSCISSA VALUE TO BE USED IN THE PLOT OF G(T).
C      TMIN=MINIMUM ABSCISSA VALUE TO BE USED IN THE PLOT OF G(T).
C      IPUNCH=0 IF THE PLOTTED POINTS OF GR(T) AND GI(T) AND THE
C      APPROXIMATE VALUES OF GHATR(T) AND GHATI(T) ARE NOT TO BE PUNCHED
C      ON DATA CARDS (ACCORDING TO FORMAT NUMBER 7 BELOW).
C      ALL COMPUTATION IS IN DOUBLE PRECISON. IF MC, THE NUMBER OF
C      TERMS IN THE EXPANSION, IS GREATER THAN 20, THE PROGRAM DIMENSION
C      STATEMENTS MUST BE ADJUSTED. ALL SUBSEQUENT SUBROUTINES ARE
C      AUTOMATICALLY CALLED ONCE THE MAIN ROUTINE APRX IS ENTERED.
C
C
       DOUBLE PRECISION CFR(20),CFI(20),GHATR,GHATI,T,SIG,BET
      .,ERRORR,RMSE,AA,BB,S2P,ESIGT,PI,
      .LSR(20,20),LSI(20,20),S1(20),S2(20)
       COMMON /COEFF/ CFR,CFI
       COMMON /MOMENT/S1,S2
       COMMON /LSCOM/LSR,LSI
       COMMON SIG,BET,MC,NNN,AA,BB,S2P,ESIGT,T,GHATR,GHATI
       EXTERNAL ERRORR
       REAL GRAPH(1000),ORDR(51),ORDI(51),ABSC(51),H(51),HH(51)
       DATA PI/3.14159265358979300/
       S2P=DSQRT(2.D0*PI)
     1 READ(5,8) MC,SIG,BET,AA,BB,GRMAX,GRMIN,TMAX,TMIN,IPUNCH
     8 FORMAT(I3,2D16.1, 2D5.1,4F5.1,I5)
       WRITE(6,23) SIG,BET,MC,AA,BB
    23 FORMAT(1H1,6HSIGMA=,D24.16//6H BETA=,D24.16//8H FOR M= ,I2,6H TERM
      .S //9H FROM T= ,D24.16//7H TO T=  ,D24.16//)
       CALL LAMBDA
       WRITE(6,18)
    18 FORMAT(//27H THE MATRIX OF LAMBDA(M,N)    //)
```

```
      DO 19 M=1,MC
   19 WRITE(6,20) (M,N,LSR(M,N),LSI(M,N),N=1,M)
   20 FORMAT(3H L(,I2,1H,I2,2H)=,D24.16,4H +I ,D24.16)
      WRITE(6,17)
   17 FORMAT(//18H THE MOMENT VECTOR      //)
      CALL COEF
      WRITE(6,16) (I,S1(I),S2(I),I=1,MC)
   16 FORMAT(3H G(,I2,2H)=,D24.16,4H +I ,D24.16   )
      WRITE(6,900)
  900 FORMAT(///70H  THE REAL AND IMAGINARY PARTS OF THE EXPANSION COEFF
     .ICIENTS , A(M)                          ///)
      WRITE(6,901) (I,CFR(I),CFI(I),I=1,MC)
  901 FORMAT(4H  A(,I2,2H)=,D24.16,5H +J  ,D24.16)
      DIV=(TMAX-TMIN)/50.
      DO 10 I=1,51
      ABSC(I)=AA+FLOAT(I-1)*DIV
      T=DBLE(ABSC(I))
      H(I)=GR(T)
      HH(I)=GI(T)
      CALL GHAT
      ORDR(I)=GHATR
      ORDI(I)=GHATI
      IF(IPUNCH) 11,10,11
   11 WRITE(7,7) ABSC(I),H(I),HH(I),ORDR(I),ORDI(I)
    7 FORMAT(5F10.4)
   10 CONTINUE
      CALL INTGRL(AA,BB,ERRORR,RMSE)
      RMSER=DSQRT(RMSE)/(BB-AA)
      CALL PLOT2(GRAPH,TMAX,TMIN,GRMAX,GRMIN)
      CALL PLOT3(1H+,ABSC(1),H(1),51)
      CALL PLOT3(1H*,ABSC(1),ORDR(1),51)
      WRITE(6,808)
  808 FORMAT(///)
      WRITE(6,800) (ABSC(I),ORDR(I),ORDI(I),H(I),HH(I),I=1,51 )
  800 FORMAT(4H  T=,F6.2,3X,9HRE(GHAT)=,E17.8,3X,9HIM(GHAT)=,E17.8,3X,
     .6HRE(G)=,E17.8,3X,6HIM(G)=,E17.8 )
      WRITE(6,812) RMSER
  812 FORMAT(////15H THE RMS ERROR=  , E10.3/)
      WRITE(6,809 )
  809 FORMAT(1H1)
      CALL PLOT4(1,1H )
      WRITE(6,801)
  801 FORMAT(35H0+=RE(G(T))    *=RE(GHAT(T))               )
      CALL PLOT2(GRAPH,TMAX,TMIN,GRMAX,GRMIN)
      CALL PLOT3(1H+,ABSC(1),HH(1),51)
      CALL PLOT3(1H*,ABSC(1),ORDI(1),51)
      WRITE(6,809)
      CALL PLOT4(1,1H  )
      WRITE(6,802)
  802 FORMAT(35H0+=IM(G(T))    *=IM(GHAT(T))               )
      GO TO 1
      CALL EXIT
      END
```

```
$IBFTC LAMBDA
C
C
C       THIS SUBROUTINE COMPUTES THE COEFFICENTS LAMBDA(M,N) IN DOUBLE
C       PRECISON. THE REAL AND IMAGINARY PARTS OF LAMBDA(M,N) ARE STORED
C       IN ARRAYS LSR(M,N) AND LSI(M,N) RESPECTIVELY.
C
C
        SUBROUTINE LAMBDA
        DOUBLE PRECISION LSR( 20,20),LSI(20,20),SIG,BET,PI,A,B,C,AM,AN,
       .Q,E,F,G,H,AK,D
        DATA PI/3.141592653589793/
        COMMON SIG,BET,MC
        COMMON /LSCOM/LSR,LSI
        A=SIG*SIG+BET*BET
        B=SIG*SIG-BET*BET
        C=2.D0*SIG*BET
        LSR(1,1)=DSQRT(SIG/PI)
        LSI(1,1)=0.D0
        DO 1 M=1,MC
        AM=M
        E=(-1)**M
        F=C*AM
        G=B*AM
        D=DSQRT((AM+1.D0)/AM)
        DO 2 N=1,M
        AN=N
        Q=D/(A*(AM+1.D0-AN))
        LSR(M+1,N)=(LSR(M,N)*(AN*A+   G)-LSI(M,N)*F)*Q
      2 LSI(M+1,N)=(LSI(M,N)*(AN*A+   G)+LSR(M,N)*F)*Q
        H=DSQRT(AM+1.D0)
        F=E*H*LSR(1,1)
        G=0.D0
        DO 3 K=1,M
        F=F-LSR(M+1,K)
      3 G=G-LSI(M+1,K)
        LSR(M+1,M+1)=F
      1 LSI(M+1,M+1)=G
        RETURN
        END


$IBFTC COEF
C
C
C       THIS SUBROUTINE COMPUTES THE FOURIER COEFFICIENTS A(M) IN DOUBLE
C       PRECISON. THE REAL AND IMAGINARY PARTS OF THE RESULTS ARE
C       STORED IN THE ARRAYS CFR(M) AND CFI(M) RESPECTIVELY.
C
C
        SUBROUTINE COEF
        EXTERNAL RINTG,IINTG
        DOUBLE PRECISION IINTG,RINTG,SIG,BET,S2P,LSR(20,20),AA,BB
```

```
     .,LSI(20,20),CFR(20),CFI(20),S1(20),S2(20),PI,C1(20,20),C2(20,20)
     .,C3(20,20),C4(20,20),ESIGT,T
      COMMON /COEFF/CFR,CFI
      COMMON /LSCOM/LSR,LSI
      COMMON /ARRAY/C1,C2,C3,C4
      COMMON /MOMENT/S1,S2
      COMMON SIG,BET,MC,NNN,AA,BB,S2P,ESIGT,T,GHATR,GHATI
      DO 10 N=1,MC
      NNN=N
      CALL INTGRL(AA,BB,RINTG,S1(N))
      CALL INTGRL(AA,BB,IINTG,S2(N))
   10 CONTINUE
      DO 1 M=1,MC
      CFR(M)=0.D0
      CFI(M)=0.D0
      DO 5 N=1,M
      CFR(M)=CFR(M)+LSR(M,N)*S1(N)+LSI(M,N)*S2(N)
      CFI(M)=CFI(M)+LSR(M,N)*S2(N)-LSI(M,N)*S1(N)
    5 CONTINUE
      CFR(M)=S2P*CFR(M)
      CFI(M)=S2P*CFI(M)
    1 CONTINUE
      DO 2 M=1,MC
      DO 2 N=1,M
      C1(M,N)=CFR(M)*LSR(M,N)-CFI(M)*LSI(M,N)
      C2(M,N)=-CFR(M)*LSI(M,N)-CFI(M)*LSR(M,N)
      C3(M,N)=CFI(M)*LSR(M,N)+CFR(M)*LSI(M,N)
    2 C4(M,N)=-CFI(M)*LSI(M,N)+CFR(M)*LSR(M,N)
      RETURN
      END


$IBFTC GHAT
C
C
C     THIS SUBROUTINE FORMS THE REAL AND IMAGINARY PARTS OF THE
C     APPROXIMATION GHAT(T) TO THE PRESCRIBED FUNCTION G(T). ALL THE
C     COMPUTATION IS IN DOUBLE PRECISION. THE REAL AND IMAGINARY PARTS
C     OF THE APPROXIMANT ARE STORED IN GHATR AND GHATI RESPECTIVELY
C     FOR EACH VALUE OF THE INDEPENDENT VARIABLE T.
C
C
      SUBROUTINE GHAT
      DOUBLE PRECISION C1(20,20),C2(20,20),C3(20,20),C4(20,20),EE,DE
     .,GHATR,GHATI,T,SIG,BET,AA,BB,S2P,ESIGT,BT,CC,SS
      COMMON /ARRAY/C1,C2,C3,C4
      COMMON SIG,BET,MC,NNN,AA,BB,S2P,ESIGT,T,GHATR,GHATI
      BT=BET*T
      ESIGT=DEXP(-SIG*T)
      GHATR=0.D0
      GHATI=0.D0
      DO 1 M=1,MC
      DE=1.D0
      DO 1 N=1,M
```

```
      DE=DE*ESIGT
      EE=BT*DBLE(FLOAT(N))
      CC=DCOS(EE)
      SS=DSIN(EE)
      GHATR=GHATR+DE*(CC*C1(M,N)+SS*C2(M,N))
    1 GHATI=GHATI+DE*(CC*C3(M,N)+SS*C4(M,N))
      GHATR=GHATR*S2P
      GHATI=GHATI*S2P
      RETURN
      END


$IBFTC RINTG
C
C
C     THIS SUBROUTINE COMPUTES THE REAL PART OF THE MOMENTS OF THE
C     PRESCRIBED FUNCTION G(T). THESE ARE USED IN COMPUTING THE
C     FOURIER EXPANSION COEFFICIENTS. COMPUTATION IS IN DOUBLE PRECISON.
C
C
      DOUBLE PRECISION FUNCTION RINTG(T)
      DOUBLE PRECISION GR,GI,SIG,BET,AN,T
      COMMON SIG,BET,MC,N
      AN=N
      RINTG=DEXP(-AN*SIG*T)*(GR(T)*DCOS(AN*BET*T)+GI(T)*DSIN(AN*BET*T))
      RETURN
      END


$IBFTC IINTG
C
C
C     THIS ROUTINE COMPUTES THE IMAGINARY PART OF THE MOMENTS OF THE
C     PRESCRIBED FUNCTION G(T). THESE ARE USED IN COMPUTING THE
C     FOURIER EXPANSION COEFFICIENTS. COMPUTATION IS IN DOUBLE PRECISON.
C
C
      DOUBLE PRECISION FUNCTION IINTG(T)
      DOUBLE PRECISION GR,GI,SIG,BET,AN,T
      COMMON SIG,BET,MC,N
      AN=N
      IINTG=DEXP(-AN*SIG*T)*(-GR(T)*DSIN(AN*BET*T)+GI(T)*DCOS(AN*BET*T))
      RETURN
      END


$IBFTC INTGRL
C
C
C     THIS PROGRAM PERFORMS INTEGRATION IN DOUBLE PRECISION AND
C     IS BASED ON THE 64 POINT GAUSSIAN QUADRATURE FORMULA.
C     A=LOWER LIMIT OF INTEGRATION IN DOUBLE PRECISION.
```

```
C      B=UPPER LIMIT OF INTEGRATION IN DOUBLE PRECISION.
C      F=THE NAME OF THE DOUBLE PRECISION FUNCTION TO BE INTEGRATED.
C      ANS=THE RESULTANT INTEGRATION IN DOUBLE PRECISION.
C
C
       SUBROUTINE INTGRL(A,B,F,ANS)
       EXTERNAL F
       DOUBLE PRECISION A,B,F,ANS,X(32),W(32),S1,S2,U(28),V(4),Y(28),Z(4)
       DATA U/.02435029266342443250 9,.07299312178779903945 0,.121462819296
      1120554470,.16964442042399281803 7,.21742364374000708415 0,.264687162
      2208767416374,.311322871990210956158,.35722015833766811595 0,.402270
      3157963991603696,.44636601725346408798 5,.48940314570705295747 9,.531
      4279464019894545658,.57189564620263403428 4,.61115535517239325024 9,.
      564896547125465733985 8,.68523631305423324256 4,.71988185017161082684
      69,.75281990726053189661 2,.78397235894334140761 0,.81326531512279755
      79742,.84062929625258036275 2,.86599939815409281976 1,.88931544599511
      84105853,.91052213707850280575 6,.92956917213193957582 1,.94641137485
      98402816062,.96100879965205371891 9,.97332682778991096374 2/
       DATA V/                                                  .98333625
      103884625956931,.99101337147674432073 9,.99634011677195527934 7,.9993
      1105041735772139457/
       DATA Y/.04869095700913972038 3,.04857546744150342693 5,.048344762234
      1802957170,.04799938859645830772 8,.04754016571483030866 2,.046968182
      2816210017325,.04628479658131441729 6,.04549162792741814448 0,.044590
      3558163756563060,.04358372452932345337 7,.04247351512365358900 7,.041
      4262563242623528610,.03995374113272034138 7,.03855015317861562912 9,.
      503705512854024004604 0,.03547221325688238381 1,.03380516183714160939
      62,.03205792835485155358 5,.03023465707240247886 8,.02833967261425948
      73228,.02637746971505465867 2,.02435270256871087333 8,.02227017380838
      83254159,.02013482315353020937 2,.01795171577569734308 5,.01572603047
      96024719322,.01346304789671864259 8,.01116813946013112881 9/
       DATA Z/                                                  .00884675
      109826363947723,.00650445796897836285 6,.00414703326056246763 5,.0017
      1183280721696432947/
       DATA ISTART/+1/
       S1=(B-A)/2.D0
       S2=(B+A)/2.D0
       IF(ISTART) 4,4,5
     5 DO 2 I=1,28
       X(I)=U(I)
     2 W(I)=Y(I)
       DO 3 I=1,4
       X(I+28)=V(I)
     3 W(I+28)=Z(I)
     4 ANS=0.D0
       DO 1 I=1,32
     1 ANS=ANS+W(I)*(F(S1*X(I)+S2)+F(-S1*X(I)+S2))
       ANS=ANS*S1
       ISTART=-0
       RETURN
       END


$IBFTC ERRORR
```

```
C
C
C     THIS ROUTINE COMPUTES THE INTEGRAL SQUARE ERROR BETWEEN THE
C     PRESCRIBED FUNCTION AND THE APPROXIMANT GENERATED BY THE
C     ACCOMPANYING PROGRAMS.  COMPUTATION IS IN DOUBLE PRECISION.
C
C
      DOUBLE PRECISION FUNCTION ERRORR(X)
      DOUBLE PRECISION SIG,BET,AA,BB,ESIGT,S2P,T,X,GHATR,GHATI,GR,GI
      COMMON SIG,BET,MC,NNN,AA,BB,S2P,ESIGT,T,GHATR,GHATI
      T=X
      CALL GHAT
      ERRORR=(GHATR-GR(X))**2+(GHATI-GI(X))**2
      RETURN
      END


      DOUBLE PRECISION FUNCTION XR(M,T)
$IBFTC XR
C
C
C     THIS ROUTINE COMPUTES THE REAL PART OF THE M TH ORTHONORMAL
C     BASIS FUNCTION X(M,T). COMPUTATION IS IN DOUBLE PRECISION.
C     THIS PROGRAM IS USED ONLY FOR CHECKING ORTHONORMALITY AND IS
C     NOT OTHERWISE USED BY THE ACCOMPANYING PROGRAMS.
C
C
      DOUBLE PRECISION  LSR(20,20),LSI(20,20),S2P,AN,SIG,BET,T,PI
     .,ESIGT,DE,EE,AA,BB
      COMMON /LSCOM/LSR,LSI
      COMMON SIG,BET,MC,NNN,AA,BB,S2P,ESIGT
      DATA PI/3.14159265358979З/
      ESIGT=DEXP(-SIG*T)
      DE=1.D0
      XR=0.D0
      DO 1 N=1,M
      AN=N
      DE=DE*ESIGT
      EE=AN*BET*T
    1 XR=XR+DE*(LSR(M,N)*DCOS(EE)-LSI(M,N)*DSIN(EE))
      XR=S2P*XR
      RETURN
      END


$IBFTC XI
C
C
C     THIS ROUTINE COMPUTES THE IMAGINARY PART OF THE M TH ORTHONORMAL
C     BASIS FUNCTION X(M,T). COMPUTATION IS IN DOUBLE PRECISION.
C     THIS PROGRAM IS USED ONLY FOR CHECKING ORTHONORMALITY AND IS
C     NOT OTHERWISE USED BY THE ACCOMPANYING PROGRAMS.
C
```

```
C
      DOUBLE PRECISION FUNCTION XI(M,T)
      DOUBLE PRECISION  LSR(20,20),LSI(20 20),S2P,AN,SIG,BET,T,PI
     .,ESIGT,DE,EE,AA,BB
      COMMON /LSCOM/LSR,LSI
      COMMON SIG,BET,MC,NNN,AA,BB,S2P,ESIGT
      ESIGT=DEXP(-SIG*T)
      DE=1.D0
      XI=0.D0
      DO 1 N=1,M
      AN=N
      EE=AN*BET*T
      DE=DE*ESIGT
    1 XI=XI+DE*(LSR(M,N)*DSIN(EE)+LSI(M,N)*DCOS(EE))
      XI=S2P*XI
      RETURN
      END


$IBFTC GR
C
C
C     THIS ROUTINE SUPPLIES THE REAL PART OF A SAMPLE FUNCTION G(T).
C     T=THE INDEPENDENT VARIABLE IN DOUBLE PRECISION.
C     GR=THE REAL PART OF G(T) IN DOUBLE PRECISION.
C
C
      DOUBLE PRECISION FUNCTION GR(T)
      DOUBLE PRECISION T
      IF(T.LE.0.D0) GO TO 1
      IF(T.LE..5D0) GO TO 2
      IF(T.LE..95D0) GO TO 3
      GR=DEXP(-2.42377D0*T)
      GO TO 10
    1 GR=0.D0
      GO TO 10
    2 GR=2.D0*T
      GO TO 10
    3 GR=-2.D0*(T-1.D0)
   10 RETURN
      END


$IBFTC GI
C
C
C     THIS ROUTINE SUPPLIES THE IMAGINARGY PART OF A SAMPLE FUNCTION G(T).
C     T=THE INDEPENDENT VARIABLE IN DOUBLE PRECISION.
C     GI=THE IMAGINARY PART OF G(T) IN DOUBLE PRECISION.
C
C
      DOUBLE PRECISION FUNCTION GI(T)
      DOUBLE PRECISION T
```

```
      GI=0.DO
      RETURN
      END


$IBMAP PLOT
C     THIS PLOTTING ROUTINE IS AN ADAPTATION OF SHARE UMPLOT FOR THE
C     IBM7044 . THE ROUTINE UTILIZES  THE FORTRAN-MAP INPUT-OUTPUT
C     P^OGRAM FACILITY WHICH IS SUPPLIED AS A SEPARATE SUBROUTINE
*                                   WRITES GRAPH IMAGES ON OUTPUT UNIT  G
*      CALLING SEQUENCES ARE
*              CALL  PLOT1(NSCALE,NH,SBH,NV,SBV)
*              CALL  PLOT2(IMAGE,XMAX,XMIN,YMAX,YMIN)
*              CALL  PLOT3(BCD,X,Y,NDATA)
*              CALL  PLOT4(NCHAR,NHABCDEF...)                         HJS
*              CALL  OMIT(ARG)                                        HJS
*              CALL  PLTAPE(ITAPE)                                    HJS
       ENTRY   PLOT1                                                    *
       ENTRY   PLOT2                                                    *
       ENTRY   PLOT3                                                    *
       ENTRY   PLOT4                                                    *
       ENTRY   FPLOT4                                                   G
       ENTRY   OMIT                                                     *
       ENTRY   PLTAPE                                                   G
*
       EXTERN  WDATA                                                    G
*
*
COL    EQU     132                    COLUMNS IN OUTPUT LINE  (1401)
SPACS  EQU 6                     SET UNUSED SPACES AT RIGHT EDGE OF PAGE
       REM                       OR CARD.  SPACS MUST BE AT LEAST 6
G      EQU COL-11-SPACS COLUMNS IN OUTPUT LINE AVAILABLE FOR IMAGE
       SPACE   5                                                        G
*      PLOT1
       REM         MAIN JOB OF PLOT1 IS TO EXAMINE ARGUMENTS AND PREPARE
       REM         SAMPLE GRIDLINE (DASH TO DASH-WORDS+1) AND SAMPLE
       REM         NON-GRID LINE (BLANK TO BLANK-WORDS+1) FOR PLOT2
*                                                                       G
PLOT1  SAVE    1,2                    ENTRY TO PLOT1                    G
*                                                                       G
       CLA     3,4        SCALE FACTORS AND DECIMAL POINT POSITIONS
       STA DELTA
       ADD     FIVE
       STA DELT
       STZ WRON1         WRON1 = 0      CLEAR ERROR FLAG, PLOT1
       CLA ONE
       STO WRON3         WRON3 = 1      SET ERROR FLAG, MISSING PLOT2
       CLA*    4,4        NH, NUMBER OF HORIZONTAL GRID LINES
       TSX FIX,2
       TZE ERR1          ZERO ARGUMENT ILLEGAL, ERROR RETURN
       STO NH
       CLA*    5,4        SBH, NO. OF SPACES BETWEEN HORIZ. GRID LINES
       TSX FIX,2
       TZE ERR1          ZERO ARGUMENT ILLEGAL, ERROR RETURN
```

```
        STO SBH
        LDQ SBH
        MPY NH
        STQ LINES          LINES = NH*SBH       MAXIMUM LINE INDEX
        CLA*     6,4        NV, NUMBER OF VERTICAL GRID LINES
        TSX FIX,2
        TZE ERR1           ZERO ARGUMENT ILLEGAL, ERROR RETURN
        STO NV
        CLA*     7,4        SBV, NO. OF SPACES BETWEEN VERT. GRID LINES
        TSX FIX,2
        TZE ERR1           ZERO ARGUMENT ILLEGAL, ERROR RETURN
        STO SBV
        LDQ SBV
        MPY NV
        STQ TOT            TOT = SBV*NV         MAXIMUM COLUMN INDEX
        CLA      TOT                                                      G
        ADD ONE
        STO TOTAL          TOTAL = TOT + 1      TOTAL COLUMNS PER LINE
        SUB GWID           WHENEVER TOTAL .G. GRAPH WIDTH, ERROR RETURN
        TMI PASS
*                                                                         G
*                                     RETURN 1, UNSUCCESSFUL PLOT1        G
*                                     WRON1=1, SET ERROR FLAG, PLOT1      G
ERR1    CAL      OTAPE                UNSUCCESSFUL PLOT1                  G
        CALL     WDATA                                                    G
        PZE      FORM                                                     G
        PZE      ER1,0,1                                                  G
        PZE      WRONG,0,3                                                G
        PZE      0                                                        G
        CLA      FPONE                                                    G
        STO      WRON1                                                    G
        RETURN   PLOT1
*                                                                         G
PASS    CLA TOTAL
        TSX FLOAT,2
        FDP SIXF
        STQ TEMP
        CLA TEMP
        FAD N999
        TSX FIX1,2
        STO WORDS          WORDS = TOTAL/6. ROUNDED UP TO NEAREST INTEGER
        LDQ WORDS          WORDS, NUMBER OF MACHINE LOCATIONS PER LINE
        MPY SIX
        STQ TOTLS          TOTLS = WORDS*6      BCD CHARACTERS PER LINE
        LXA WORDS,2
        CLA TOTLS
        SUB TOTAL
        PAX 0,1
        CLA DSH,1          LAST WORD OF A HORIZONTAL GRID LINE
        STO DASH+1,2       SET UP LAST WORD IN HORIZONTAL GRID LINE IMAGE
        LDQ BLNKK          LAST WORD OF NON GRID LINE
        STQ BLANK+1,2      SET UP LAST WORD IN NON GRID LINE IMAGE
        TIX      GA1,2,1                                                  G
        TRA      GA2                  ONE WORD PER LINE CASE              G
*                                                                         G
```

```
GA1     CLA     DSH                                                            G
GA3     STO     DASH+1,2               SET REMAINDER OF HORIZ. GRID            G
        STQ BLANK+1,2    SET UP REMAINDER OF NON GRID LINE IMAGE
        TIX     GA3,2,1                                                        G
*                                                                              G
GA2     STZ     I                      COL. INDEX FOR VERTICAL GRID            G
*                                                                              G
GAMMA   TSX     PLACB,4                PUT VERTICAL GRID I IN IMAGE            G
        PZE     IEYE
        PZE     I
        PZE     BLANK
*                                                                              G
        TSX     PLACB,4                PUT + AT INTERSECTIONS                  G
        PZE     IPLUS
        PZE     I
        PZE     DASH
*                                                                              G
        CLA I
        ADD SBV
        STO I            I=I+SBV, INCREMENT COLUMN INDEX FOR VERT GRID
        SUB TOTAL
        TZE GAMMA        IF ZERO OR MINUS LINE IS UNFINISHED, RETURN
        TMI GAMMA
*                                                                              G
DELTA   CLA **  NSCALE   NSCALE, DETERMINES SCALE FACTOR MODIFICATION
        TZE ETA          STANDARD SCALE FACTORS AND DEC POINT POSITIONS
*                                                                              G
        AXT     4,4       G7= DEC POINT POSITION FOR X
DELT    CLA     **,4  NSCALE+5       G5 = SCALE FACTOR FOR X                   G
        TSX     TFIX,2    G4= DEC POINT POSITION FOR Y
        STO     G3+4,4                 G3 = SCALE FACTOR FOR Y                 G
        TIX     DELT,4,1
*                                                                              G
ETA     CLA G4
        TZE     GA7                                                            G
        TPL     GA7                                                            G
        ZAC                            NEG.DEC.PT. POSITION = 0                G
        TRA     GA8                                                            G
GA7     CAS     EIGHT                  MAX. DEC. PLACES, ORDINATE, =8          G
        CLA     EIGHT     IF G4 GTR THAN 8, SET G4=8
        NOP                                                                    G
GA8     STO     G4                                                             G
        CLA     G7                                                             G
GA9     SUB     TEN                    ABSCISSA DEC. POINT IS MOD 10           G
        TPL     GA9                                                            G
        ADD TEN
        TZE     GA10                                                           G
        TPL     GA10                                                           G
        ZAC                            NEG.DEC.PT. POSITION = 0                G
GA10    STO     G7                                                             G
        CLA SBV          SBV, COLUMNS AVAILABLE FOR EACH ABSCISSA VALUE
        STO G9           G9 = SBV        FIELD WIDTH FOR ABSCISSA VALUES G
        CAS     G7        IF G7 GTR THAN OR EQU TO G9, SET G7=G9-1
        TRA     PASS4     ENSURES DEC POINT INSIDE FIELD
        NOP
```

```
         SUB ONE
         STO      G7                    G9-1                         G
PASS4    CLA TWELV         TWELVE SPACES ON LEFT FOR ORDINATE AND LABEL
         ADD G7
         STO G6            G6 = G7+12     FIELD WIDTH, LEFT ABSCISSA VALUE
         CLA RECLT
         SUB TOTAL
         SUB G6
         TPL      PASS5                 RECLT-TOTAL-G6.LT.O, REDUCE G6       G
         ADD G6
         STO G6            G6  REDUCED TO NUMBER OF COLUMNS AVAILABLE
PASS5    CLA G6            WHENEVER G7.GE.G6, G7 = G6-1
         CAS      G7        ENSURES DEC POINT INSIDE LEFTMOST FIELD
         TRA      EXIT
         NOP
         SUB      ONE
         STO      G7                    G6-1                         G
*                                                                    G
*                                                                    G
*                                       SET THE FORMATS              G
*                                                                    G
EXIT     LDQ      G3                                               HJS
         TSL      BCDCON                                           HJS
         SLW      FM1A                                             HJS
         CLA      G3       IF G3 IS NEGATIVE SET SCALE             HJS
         TZE      HJS1     FACTOR IN FM1 TO NEGATIVE               HJS
         TPL      HJS1                                             HJS
         MSM      FM1A                                             HJS
HJS1     LDQ      G4                                               HJS
         TSL      BCDCON                                             G
         SLW      FM1B                                               G
         LDQ      G5                                               HJS
         TSL      BCDCON                                           HJS
         SLW      FM3A                                             HJS
         CLA      G5       IF G5 IS NEGATIVE SET SCALE             HJS
         TZE      HJS2     FACTOR IN FM3 TO NEGATIVE               HJS
         TPL      HJS2                                             HJS
         MSM      FM3A                                             HJS
HJS2     LDQ      G6                                               HJS
         TSL      BCDCON                                             G
         SLW      FM3B                                               G
         LDQ      G7                                                 G
         TSL      BCDCON                                             G
         SLW      FM3C                                               G
         SLW      FM3F                                               G
         LDQ      NV                                                 G
         TSL      BCDCON                                             G
         SLW      FM3D                                               G
         LDQ      G9                                                 G
         TSL      BCDCON                                             G
         SLW      FM3E                                               G
*                                                                    G
         ZAC                                                         G
         RETURN   PLOT1     EXIT, SUCCESSFUL PLOT1
         SPACE    5                                                  G
```

```
*       PLOT2                                                              HJS
        REM          MAIN JOB OF PLOT2 IS TO REPEATEDLY LAY DOWN SAMPLE
        REM          GRIDLINE (DASH TO DASH-WORDS+1) FOLLOWED BY (SBH-1)
        REM          NON-GRID LINES (BLANK TO BLANK-WORDS+1) TO FORM THE
        REM          GRID IN THE IMAGE REGION
*                                                                            G
*                                                                            G
PLOT2   SAVE    1,2                    ENTRY TO PLOT2                        G
*                                                                            G
        STZ WRON3        WRON3 = 0       CLEAR ERROR FLAG, MISSING PLOT2
        STZ     WRON2                  CLEAR ERROR FLAG, PLOT2               G
        CLA WRON1
        TNZ     GA12                                                         G
        CLA     3,4                    IMAGE ADDRESS                         G
        STA     PLT22                                                        G
        STA     PLT23                                                        G
        STA PLT37
        CLA*    4,4                    XMAX, MAX. ABSCISSA VALUE             G
        TSX TSTFP,2      TEST FOR FLOATING POINT ARGUMENT
        TRA     BAD                                                          G
        STO XMAX
        CLA*    5,4                    XMIN, MIN. ABSCISSA VALUE             G
        TSX TSTFP,2      TEST FOR FLOATING POINT ARGUMENT
        TRA     BAD                                                          G
        STO XMIN
        CLA*    6,4                    YMAX, MAX. ORDINATE VALUE             G
        TSX TSTFP,2      TEST FOR FLOATING POINT ARGUMENT
        TRA     BAD                                                          G
        STO YMAX
        CLA*    7,4                    YMIN, MIN. ORDINATE VALUE             G
        TSX TSTFP,2      TEST FOR FLOATING POINT ARGUMENT
        TRA     BAD                                                          G
        STO YMIN
*                                                                            G
        CLS XMIN
        FAD XMAX
        TZE     BAD                    ERROR IF XMIN .EQ. XMAX               G
        STO SPANX        SPANX = XMAX - XMIN      ABSCISSA SPAN
*                                                                            G
        CLS YMIN
        FAD YMAX
        TZE     BAD                    ERROR IF YMIN .EQ. YMAX               G
        STO SPANY        SPANY = YMAX - YMIN      ORDINATE SPAN
*                                                                            G
        CLA LINES
        TSX FLOAT,2
        FDP SPANY
        STQ U            U = LINES/SPANY     NUMBER OF LINES PER UNIT Y
*                                                                            G
        CLA TOT
        TSX FLOAT,2
        FDP SPANX
        STQ V            V = TOT/SPANX       NUMBER OF COLUMNS PER UNIT X
*                                                                            G
        CLA NV
```

```
        TSX FLOAT,2
        STO TEMP
        CLA SPANX
        FDP TEMP
        STQ DELTX         DETLX = SPANX/NV    X INCR BETWN VERT GRID LINES
*                                                                         G
        CLA NH
        TSX FLOAT,2
        STO TEMP
        CLA SPANY
        FDP TEMP
        STQ DELTY         DELTY = SPANY/NH    Y INCR BETWN HORZ GRID LINES
*                                                                         G
        STZ I             I=0  INITIALIZE WORD COUNTER FOR IMAGE REGION
        STZ J             J=0  INITIALIZE LINE COUNTER FOR IMAGE REGION
*                                                                         G
*                                                                         G
TNEXT   STZ K             K=0  INITIALIZE WORD COUNTER FOR HORZ GRID LINE
*                                                                         G
LNEXT   CLA     K                     LOOP TO PLACE ONE HORIZONTAL        G
        PAX 0,2           GRID LINE IMAGE (DASH REGION) INTO IMAGE REGION
        ADD I
        PAC     0,4                                                       G
        CLA DASH,2
PLT22   STO     **,4  IMAGE                                               G
        CLA K
        ADD ONE
        STO K             K = K+1       INCREMENT WORD COUNTER FOR LINE
        SUB WORDS
        TNZ     LNEXT                 IF NON-ZERO, LINE NOT FINISHED      G
*                                                                         G
*                                                                         G
        CLA I
        ADD WORDS
        STO I             I = I+WORDS    INCR WORD COUNTER FOR NEW LINE
*                                                                         G
        CLA     NH                    SEE IF FINISHED                     G
        SUB J
        TZE     GA13                  WHEN J.EQ.NH, IMAGE GRID COMPLETE   G
*                                                                         G
        CLA ONE
        STO TEMP          TEMP = 1  INITIALIZE BETWEEN GRID LINE COUNTER
*                                                                         G
TFIN    STZ K             K=0  INITIALIZE WORD COUNTER FOR EACH LINE
*                                                                         G
LFIN    CLA     K                     LOOP TO PLACE SBH NON-GRID LINE     G
        PAX 0,2           IMAGES (BLANK REGION) INTO THE IMAGE REGION
        ADD I
        PAC     0,4                                                       G
        CLA BLANK,2
PLT23   STO     **,4  IMAGE                                               G
        CLA K
        ADD ONE
        STO K             K = K+1       INCREMENT WORD COUNTER FOR LINE
        SUB WORDS
```

```
       TNZ      LFIN                      IF NON-ZERO, LINE NOT FINISHED      G
*                                                                             G
       CLA I
       ADD WORDS
       STO I              I = I+WORDS    INCR WORD COUNTER FOR NEW LINE
       CLA TEMP
       ADD ONE
       STO TEMP           TEMP = TEMP+1  INCREMENT BETWN GRID LINE COUNTER
       SUB SBH
       TNZ      TFIN                      IF NON-ZERO, MORE LINES REQUIRED    G
*                                                                             G
       CLA J
       ADD ONE
       STO J              J = J+1   INCREMENT LINE COUNT FOR IMAGE REGION
       TRA      TNEXT                     RETURN FOR ANOTHER HORIZ. GRID LINE G
*                                                                             G
*                                                                             G
*                                         RETURN 2, UNSUCCESSFUL PLOT2        G
*                                         SET ERROR FLAG, PLOT?               G
BAD    CAL      CTAPE                                                         G
       CALL     WDATA                                                         G
       PZE      FORM                                                          G
       PZE      ER2,0,1                                                       G
       PZE      WRONG,0,3                                                     G
       PZE      0                                                             G
*                                                                             G
GA12   CLA      ONE                       SET 'NO PLOT2' FLAG                 G
       STO      WRON3                                                         G
       CLA      FPTWO                                                         G
       STO      WRON2                                                         G
*                                                                             G
*                                                                             G
GA13   RETURN   PLOT2                                                         G
       SPACE    5                                                           HJS
*      PLOT3
       REM          PLOT3 EXAMINES THE DATA POINT TO MAKE SURE IT IS
       REM          FLOATING POINT AND THEN PLACES IT IN THE PROPER SPOT
       REM          IN THE IMAGE REGION
*                                                                             G
PLOT3  SAVE     1,2                       PLOT3 ENTRY POINT                   G
*                                                                             G
       STZ FLAG1          FLAG1 = 0       PLOT3 RETURN PRESET TO ZERO
       CLA      3,4                       ADDRESS, PLOTTING CHARACTER         G
       STA PLT36
       CLA      4,4                       BASE ADDRESS, X COORDINATES         G
       STA PLT35
       CLA      5,4                       BASE ADDRESS, Y COORDINATES         G
       STA PLT34
*                                                                             G
       CLA WRON1
       ORA WRON2
       TNZ      GA16                      OUT IF BAD PLOT1 OR PLOT2           G
*                                                                             G
*                                                                             G
       ORA      WRON3                                                         G
```

```
        TZE     GA14                    OUT IF PREVIOUS PLOT2                G
*                                                                            G
        CAL     OTAPE                   PLOT3 W/O PLOT2                      G
        CALL    NDATA                                                        G
        PZE     FORM                                                         G
        PZE     ER3,0,3                                                      G
        PZE     0                                                            G
GA16    CLA     FTHRE                                                        G
*                                                                            G
GA18    RETURN  PLOT3                                                        G
        SPACE   5                                                            G
*                                                                            G
GA14    CLA*    6,4                     NDATA, NO. OF POINTS                 G
        TSX FIX,2
        TNZ     GA17                                                         G
*                                                                            G
        CLS FTHRE          IF NDATA = 0, NO DATA POINTS. RETURN MINUS THREE
        TRA     GA18                                                         G
*                                                                            G
GA17    STO     NDATA                   NO. OF POINTS                        G
        STZ K              K=0        INITIALIZE DATA POINT COUNTER
*                                                                            G
LTEND   LAC     K,1                                                          G
*                                                                            G
PLT34   CLS **,1           Y(K)       Y COORDINATE OF (K+1)TH DATA POINT
        TSX TSTFP,2        TEST FOR FLOATING POINT ARGUMENT
        TRA     GA21                                                         G
        FAD YMAX
        LRS     35
        FMP U
        TPL     GA19                                                         G
        FSB N05
        TRA     GA20                                                         G
GA19    FAD     N05                                                          G
GA20    TSX     FIX1,2                                                       G
        STO I              I=(YMAX-Y(K))*U +OR- 0.5, LINE INDEX FOR DATA PT
        TZE     PLT35                   Y LIES ON TOP GRID LINE              G
        TMI     GA21                    REJECT Y IF ABOVE TOP GRID LINE      G
        SUB LINES
        TZE     PLT35                   Y LIES ON BOTTOM GRID LINE           G
        TPL     GA21                    REJECT Y IF BELOW BOTTOM GRID LINE   G
*                                                                            G
PLT35   CLA **,1           X(K)       X COORDINATE OF (K+1)TH DATA POINT
        TSX TSTFP,2        TEST FOR FLOATING POINT ARGUMENT
        TRA     GA21                                                         G
        FSB XMIN
        LRS     35
        FMP V
        TPL     GA22                                                         G
        FSB N05
        TRA     GA23                                                         G
GA22    FAD     N05                                                          G
GA23    TSX     FIX1,2                                                       G
        STO J              J=(X(K)-XMIN)*V +OR- 0.5, COL INDEX FOR DATA PT
        TZE     GA24                    X LIES ON LEFTMOST GRID LINE         G
```

```
        TMI     GA21                    REJECT X IF LEFT OF GRID            G
        SUB TOT
        TZE     GA24                    X LIES ON RIGHT GRID LINE           G
        TPL     GA21                    REJECT X IF RIGHT OF GRID           G
GA24    LDQ     TOTLS                                                       G
        MPY I
        LLS     35
        ADD J
        STO L              L=TOTLS*I+J, CHARACTER POSITION IN IMAGE REGION
        TSX     PLACF,4                 PLACE BCD IN L-TH                   G
PLT36   PZE     **  BCD                                                     G
        PZE     L                                                           G
PLT37   PZE     **   IMAGE                                                  G
*                                                                           G
THEND   CLA K
        ADD ONE
        STO K              K=K+1      INCREMENT DATA POINT COUNTER
        SUB NDATA
        TNZ LTEND ,        IF NONZERO, MORE DATA POINTS TO BE PLOTTED
*                                                                           G
        CLA FLAG1          PLOT3 RETURN
        RETURN  PLOT3                                                       G
*                                                                           G
GA21    CLS     FTHRE                   PLOT3 REJECTED POINT                G
        STO     FLAG1                                                       G
        TRA     THEND                                                       G
        SPACE   5                                                           G
*PLOT4 ( FPLOT4 )                                                         HJS
        REM          PLOT4 DECOMPOSES THE STRING OF CHARACTERS IN LABEL
        REM          AND WRITES THE CURRENT GRAPH ON TAPE OTAPE
*                                                                           G
*                                                                           G
PLOT4   SAVE    1,2                     ENTRY TO PLOT4                      G
*                                                                           G
*                    W034 ASSUMES ALL ARRAYS ARE STORED FORWARD           HJS
FPLOT4 SYN      PLOT4                   THEREFORE BOTH ENTRIES ARE THE SAME G
*                                                                           G
        CLA WRON1
        ORA WRON2
        TNZ     GA26                    OUT IF BAD PLOT1 OR PLOT2           G
*                                                                           G
*                                                                           G
        ORA     WRON3                                                       G
        TZE     GA27                    OUT IF PREVIOUS PLOT2               G
*                                                                           G
        CAL     OTAPE                   UNSUCCESSFUL PLOT4                  G
        CALL    WDATA                                                       G
        PZE     FORM                                                        G
        PZE     ER3,0,3                 NO PREVIOUS PLOT2                   G
        PZE     0                                                           G
*                                                                           G
GA26    CLA     FPFOR                   RETURN 4, UNSUCCESSFUL PLOT4        G
        RETURN  PLOT4                                                       G
*                                                                           G
GA27    CLA     4,4                     LABEL BASE ADDRESS                  G
```

```
        STA PLT41
        STA PLT42
        CLA     PLT37             IMAGE ADDRESS                     G
        STA     ADDB                                                G
        STA     GA49                                                G
        LXA     WORDS,2           WORDS PER LINE                    G
        SXD     ADDB,2                                              G
        SXD     GA49,2                                              G
        CLA*    3,4               NCHAR, NO. OF CHARACTERS IN LABEL G
        TSX FIX,2
        ADD ONE
        PAX     0,4               NCHAR+1                           G
        AXT     6,1               SET CHAR. COUNT FOR LABEL WORD  HJS
        CLA     YMAX              SET TOP LINE                      G
        STO     YAXIS             ORDINATE VALUE                    G
*                                                                   G
PLT41   LDQ     **  LABEL         GET FIRST LABEL WORD              G
        STQ LABEL
        CLA LINES
        STO FIXV        FIXV = LINES   MAXIMUM HORIZONTAL LINE INDEX
        CAL IFOMT
        ANA IFOUR       IF IFOMT = 4,5,6, OR 7, DELETE BOTTOM GRID LINE
        TZE     GA29                                                G
        CLA LINES
        SUB ONE
        STO FIXV        FIXV = LINES-1, MAX LINE INDEX WITH NO BOTT LINE
*                                                                   G
GA29    STZ     I                 INITIALIZE LINE COUNT FOR IMAGE   G
        AXT     0,2     INITIALIZE WORD COUNTER FOR IMAGE REGION  HJS
*                                                                   G
*                                                                   G
CHECK   CLA FIXV
        SUB I
        TMI     GA30              FIXV-I NEGATIVE, IMAGE PRINT COMPLETG
*                                                                   G
        TXH     HJS3,4,1     IF LABEL HAS BEEN COMPLETELY         HJS
        CAL     BLNKK        PRINTED, OR IF NO LABEL IS WANTED,   HJS
        SLW     LABEL        SET LABEL TO BLANK                   HJS
HJS3    ZAC                                                       HJS
        LDQ I
        DVP SBH
        TNZ SKIP        IF NON-INTEGRAL, BYPASS ORDINATE PREPARATION
        CAL IFOMT
        ANA TWO
        TNZ SKIP        IF IFOMT=2,3,OR 6, DELETE ORDINATE VALUE
*                                                                   G
        CAL     OTAPE             GRID-LINE IMAGE                   G
        CALL    WDATA                                               G
        PZE     FMI                                                 G
        PZE     LABEL,0,1         LABEL CHARACTER                   G
        PZE     YAXIS,0,1                                           G
ADDB    PZE     **,0,**  IMAGE,0,WORDS                              G
        PZE     0                                                   G
*                                                                   G
        CLA     YAXIS             ADJUST ORDINATE VALUE             G
```

```
        FSB     DELTY                                               G
        STO     YAXIS                                               G
*                                                                   G
GA50    CAL     ADDB                   MOVE TO NEXT IMAGE LINE      G
        ADD     WORDS                                               G
        STA     ADDR                                                G
        STA     GA49                                                G
        CLA     I                      COUNT LINES                  G
        ADD     ONE                                                 G
        STO     I                                                   G
*                                                                   G
        TNX     CHECK,4,1              DECREMENT LABEL CHAR. COUNT  HJS
        CAL     LABEL                  SET NEXT LABEL CHARACTER     G
        ALS     6                                                   G
        TIX     GA32,1,1 DECREMENT CHARACTER COUNTER IN LABEL WORD  HJS
        AXT     6,1      REINITIALIZE CHAR. COUNT FOR LABEL WORD    HJS
        TXI     *+1,2,-1               MOVE TO AND                  G
PLT42   CAL     **,2  LABEL            GET NEXT LABEL WORD          G
GA32    SLW     LABEL                  SAVE LABEL WORD              G
*                                                                   G
        TRA     CHECK                  AROUND FOR NEXT LINE         G
*                                                                   G
*                                                                   G
SKIP    CAL     OTAPE                  WRITE IMAGE, NON-GRID-LINE   G
        CALL    WDATA                                               G
        PZE     FM2                                                 G
        PZE     LABEL,0,1                                           G
GA49    PZE     **,0,**  IMAGE,0,WORDS                              G
        PZE     0                                                   G
*                                                                   G
        TRA     GA50                                                G
*                                                                   G
*                                                                   G
GA30    CAL     IFOMT                                               G
        ANA ONE
        TNZ EXIT2        IF IFOMT=1,3,5, OR 7, DELETE ABSCISSA PRINTOUT
*                                                                   G
        CLA     XMIN                   FIRST ABSCISSA VALUE         G
        STO     ABS                                                 G
        LXA     NV,4                   FORM ABSCISSA VALUES         G
        TXI     *+1,4,1                NV+1                         G
        SXD     GA51,4                                              G
        TXI     *+1,4,-1               NV                           G
        AXT     0,1                                                 G
        CLA     ABS,1                                               G
LS2     FAD     DELTX                                               G
        STO     ABS+1,1                                             G
        TXI     *+1,1,-1                                            G
        TIX     LS2,4,1                                             G
*                                      PUT OUT THE ABSCISSA LINE    G
        CAL     OTAPE
        CALL    WDATA                                               G
        PZE     FM3                                                 G
GA51    PZE     ABS,0,**  NV+1                                      G
        PZE     0                                                   G
```

```
*                                                                    G
EXIT2  ZAC                                                         HJS
       RETURN PLOT4                                                HJS
       SPACE    5                                                    G
*      OMIT
*
OMIT   SAVE     2                    DELETE PRINTOUT SEGMENTS        G
       CLA*     3,4                                                  G
*                                    ARGUMENT TAKEN MOD 8, AS FOLLOWS G
*                                    ARG=1, DELETE ABSCISSA VALUE PRINT G
       TSX TFIX,2          ARG = 2, DELETE ORDINATE VALUE PRINTOUT
       TMI      GA37                 ARG=3, 1 AND 2                  G
       ORA      IFOMT                ARG=4, DELETE BOTTOM GRID LINE  G
       SLW      IFOMT                                                G
       RETURN   OMIT                 ARG=5, 1 AND 4                  G
*                                                                    G
GA37   COM                           ARG=6, 2 AND 4                  G
       ANA      IFOMT                ARG=7, 1, 2, AND 4              G
       SLW      IFOMT                                                G
       RETURN   OMIT                 TO RESTORE, EXECUTE OMIT        G
       REM                 WITH THE NEGATIVES OF THE ABOVE ARGUMENTS
       SPACE    5                                                    G
*      PLTAPE
*
PLTAPE SAVE     2                    CHANGE OUTPUT TAPE NO.          G
       CLA*     3,4                                                  G
       TSX      FIX,2                                                G
       STO      OTAPE                                                G
       RETURN   PLTAPE                                               G
       SPACE    5                                                    G
*      BCDCON
*
*                                    CONVERT INTEGER IN MQ           G
*                                    TO BCD (999999 MAX.)            G
*                                    IN LOGICAL AC.                  G
*                                                                    G
*                                                                    G
BCDCON AXT      **,0                                                 G
       AXT      6,4                                                  G
       ZAC                                                           G
BCD1   SLW      BCD2                                                 G
       ZAC                                                           G
       DVP      TEN                                                  G
       XEC      BCD2,4                                               G
       ORA      BCD2                                                 G
       TIX      BCD1,4,1                                             G
       TRA*     BCDCON                                               G
       NOP                                                           G
       ALS      6                                                    G
       ALS      12                                                   G
       ALS      18                                                   G
       ALS      24                                                   G
       ALS      30                                                   G
BCD2   DEC      0                                                    G
*                                                                    G
```

```
*                                                                        G
        SPACE   5                                                        G
*       TSTFP
*
TSTFP   TZE     2,2                       CHECK FOR FLOATING POINT       G
        STO TEMP           ITS CALLING SEQUENCE IS
        SSP                     TSX TSTFP,2
        SUB MS004          THE ARGUMENT IS IN THE ACCUMULATOR
        TMI     GA11                                                     G
        CLA TEMP
        TRA     2,2                       FLOATING-POINT RETURN          G
GA11    CLA     TEMP                                                     G
        TRA     1,2                       NON-FLOATING-POINT RETURN      G
        SPACE   5                                                        G
*       FLOAT
*
FLOAT   ORA CONST          FLOAT FLOATS A NUMBER KNOWN TO BE AN INTEGER
        FAD CONST          THE CALLING SEQUENCE IS
        TRA 1,2                 TSX FLOAT,2  WITH ARGUMENT IN ACCUMULATOR
        SPACE   5                                                        G
*       FIX
*
*                                 CONVERT ARGUMENT TO FULL-WORD INTEGEG
FIX     SSP                       POSITIVE RESULT                        G
TFIX    LRS     26                SIGNED RESULT                          G
        TNZ     FIX2              IF NON-ZERO, CONSIDERED FLOATING       G
        LLS     26                ALREADY FIXED                          G
        TRA     1,2
FIX2    LLS     26                RESTORE FLOATING NUMBER                G
FIX1    UFA CONST          FIXES A NUMBER KNOWN TO BE IN FLOATING POINT
        LRS 27
        ZAC                                                              G
        LLS 27
        TRA 1,2
        SPACE   5                                                        G
*       PLACF, PLACB
*
*                                 PLACE BCD CHARACTER IN                 G
*                                 I-TH CHARACTER POSITION OF             G
*                                 A SPECIFIED REGION                     G
*                                 TSX PLACE,4                            G
*                                 PZE BCD                                G
*                                 PZE I                                  G
*                                 PZE REGION                             G
PLACF   MSM     GA36              -GA36, FORWARD ARRAYS                  G
        TRA     GA35                                                     G
*                                                                        G
PLACB   MSP     GA36              +GA36, BACKWARD ARRAYS                 G
*                                                                        G
GA35    LDQ*    2,4               I                                      G
        STQ     TEM                                                      G
        ZAC                                                              G
        DVP SIX
        STQ TEM1           TEM1=I/6, INDEX OF WORD CONTAINING CHAR POSITION
        LAC     TEM1,2            SET FOR FORWARD ARRAYS                 G
```

```
        MIT      GA36                 TEST IF FORWARD ARRAY               G
        LXA      TEM1,2               NO, SET FOR BACKWARD ARRAYS         G
        MPY      SIX                                                      G
        STQ TEM1            TEM1=TEM1*6,  CONTAINS TOTAL CHARACTER POSITIONS
        CLA TEM             TEM, CONTAINS THE CHARACTER POSITION
        SUB TEM1
        PAX 0,1             TEM-TEM1 = CHARACTER POSITION IN WORD
        CLA 3,4
        STA      GA4                                                      G
        STA      GA5                                                      G
        CAL PLCMK,1
GA4     ANA      **,2                 ZERO THE CHARACTER POSITION         G
        SLW*     GA4                                                      G
        CAL*     1,4                  GET AND                             G
        ANA      MASK                 ISOLATE THE CHARACTER               G
        TNX      GA5,1,0                                                  G
GA6     ARS      6                    SHIFT IT INTO POSITION              G
        TIX      GA6,1,1                                                  G
GA5     ORA      **,2                 PUT CHARACTER INTO WORD             G
        SLW*     GA5                                                      G
        TRA 4,4             RETURN
*                                                                         G
GA36    PZE      0                    -=FORWARD ARRAY, +=BACKWARD ARRAY   G
        SPACE    5                                                        G
ABS     BSS      COL/2                ABSCISSA VALUES                     G
        BCI      3,                 I
        BCI      1,I                  SAMPLE LINE IMAGE                   G
        BCI      1,          FOR NON-GRID LINES.
        BCI      1,    I     THIS IS LINE IMAGE USED IN STANDARD GRID.
        BCI      1,          EXECUTION OF PLOT1 SETS UP NEW VALUES.
        BCI      1,      I
        BCI      1,I
        BCI      1,
        BCI      1,    I
        BCI      1,
        BCI      1,      I
        BCI      1,I
        BCI      1,
        BCI      1,    I
        BCI      1,
        BCI      1,      I
BLANK   BCI      1,I         FIRST WORD OF LINE IMAGE FOR NON-GRID LINES
BLNKK   BCI      1,
CONST   OCT 233000000000
        BCI      3,            ----+
        BCI      1,+-----             SAMPLE LINE IMAGE                   G
        BCI      1,------    HORIZONTAL GRID LINES.
        BCI      1,---+---   THIS IS THE LINE IMAGE USED IN STANDARD GRID.
        BCI      1,------    EXECUTION OF PLOT1 PRODUCES NEW VALUES.
        BCI      1,----+-
        BCI      1,+-----
        BCI      1,------
        BCI      1,--+---
        BCI      1,------
        BCI      1,----+-
```

```
         BCI     1,+-----
         BCI     1,------
         BCI     1,--+---
         BCI     1,------
         BCI     1,----+-
DASH     BCI     1,+-----   1ST WORD OF LINE IMAGE FOR HORIZ GRID LINES
DELTX    DEC     0.                  INCR. BETWEEN VERTICAL LINES            G
DELTY    DEC     0.                  INCR. BETWEEN HORIZ. LINES              G
         BCI     1,-        DSH TO DSH-5 USED BY PLOT1 TO FILL OUT LAST
         BCI     1,--       WORD OF HORIZ LINE IMAGE
         BCI     1,---
         BCI     1,----
         BCI     1,-----
DSH      BCI     1,------
EIGHT    DEC 8
ER1      BCI     1,0PLOT1
ER2      BCI     1,0PLOT2                                                    G
ER3      BCI     3,0NO PREVIOUS PLOT2
FIVE     DEC     5
FIXV                        MAXIMUM HORIZONTAL LINE INDEX FOR PRINTING
FLAG1
*
*                                    FORMATS FOR PRINTING THE IMAGE          G
*                                                                            G
FM1      BCI     1,(1XA1,            GRID LINES                              G
FM1A     BCI     1,     0            G3                                      G
         BCI     1,PF9.                                                      G
FM1B     BCI     1,     3            G4                                      G
         BCI     2,,1X20A6)                                                  G
*                                                                            G
*                                    NON-GRID LINES                          G
FM2      BCI     3,(1XA1,10X20A6)                                            G
*                                                                            G
*                                    ABSCISSA LINE                           G
FM3      BCI     1,(1H0                                                      G
FM3A     BCI     1,     0            G5                                      G
         BCI     1,PF                                                        G
FM3B     BCI     1,    15            G6                                      G
         BCI     1,.                                                         G
FM3C     BCI     1,     3            G7                                      G
         BCI     1,,                                                         G
FM3D     BCI     1,    10            NV                                      G
         BCI     1,F                                                         G
FM3E     BCI     1,    10            G9                                      G
         BCI     1,.                                                         G
FM3F     BCI     1,     3            G7                                      G
         BCI     1,)                                                         G
FORM     BCI     1,(22A6)                                                    G
FPONE    DEC 1.
FPTWO    DEC 2.
FTHRE    DEC 3.
FPFOR    DEC 4.
         SPACE   5                                                           G
*                                    FORMAT PARAMETERS                       G
*                                    MODIFIABLE BY PLOT1                     G
```

```
*                                                                      G
*                ORDINATE              G3 P F 9.G4                     G
*                LEFTMOST ABSCISSA     G5 P F G6.G7                    G
*                OTHER ABSCISSA        G5 P F G9.G7                    G
G3      DEC      0                                                     G
G4      DEC      3                                                     G
G5      DEC      0                                                     G
G7      DEC      3                                                     G
*                                                                      G
G6      DEC      15                    G7+12                           G
G9      DEC      10                    SBV                             G
        SPACE    5                                                     G
GWID    PZE G              GRAPH WIDTH (NUMBER OF COLUMNS FOR IMAGE + 1)
I
IEYE    BCI      1,I
IPLUS   BCI      1,+
IFOMT                      SWITCH CONTAINING INFORMATION FROM OMIT
IFOUR   DEC 4
J                                                                      G
K
L
LABEL
LINES   DEC 50         *** MAXIMUM HORIZONTAL LINE INDEX
MS004   OCT 000777777777
        OCT 777777777700
        OCT 777777770077
        OCT 777777007777
        OCT 777700777777
        OCT 770077777777
PLCMK   OCT 007777777777
MASK    OCT 770000000000
N05     DEC 0.5
N999    DEC .99
NDATA                      NUMBER OF DATA POINTS TO BE PLOTTED
NH      DEC 5          *** NUMBER OF HORIZONTAL GRID LINES
NV      DEC 10         *** NUMBER OF VERTICAL GRID LINES
ONE     DEC 1
OTAPE   DEC      6                     OUTPUT TAPE (SET BY 'PLTAPE')   G
RECLT   PZE COL            NUMBER OF COLUMNS IN OUTPUT LINE
SBH     DEC 10         *** NUMBER OF SPACES BETWEEN HORIZONTAL GRIDLINES
SBV     PZE 10         *** SPACES BETWEEN VERTICAL GRID LINES          *
SIX     DEC 6
SIXF    DEC 6.
SPANX   DEC      0                     XMAX-XMIN                       G
SPANY   DEC      0                     YMAX-YMIN                       G
TEM
TEM1
TEMP
TEN     DEC 10
TOT     DEC 100        *** MAXIMUM COLUMN INDEX
TOTAL   DEC 101        *** TOTAL COLUMNS PER LINE
TOTLS   DEC 102        *** NUMBER OF BCD CHARACTERS PER LINE
TWELV   DEC 12
TWO     DEC 2
U       DEC      0.                    LINES PER UNIT Y                G
```

```
V       DEC      0.                     LINES PER UNIT X                  G
WORDS   DEC 17        *** NUMBER OF MACHINE LOCATIONS PER LINE
WRONG   BCI      3, IMPROPER ARGUMENT
WRON1   DEC 0         *** EQUALS 1 FOR UNSUCCESSFUL PLOT1
WRON2                     EQUALS 1 FOR UNSUCCESSFUL PLOT2
WRON3   DEC 1         *** EQUALS 1 UNTIL SUCCESSFUL PLOT2
YAXIS                                                                     G
XMAX
XMIN
YMAX
YMIN
        END               THIS IS THE LAST CARD


$IBMAP IO
C      THIS PROGRAM IS USED IN CONJUNCTION WITH THE PLOTTING ROUTINE
C      TO ALLOW FORTRAN INPUT-OUTPUT FACILITY IN A MACHINE LANGUAGE
C      PROGRAM
*               ---------RAND  W038   I/O ROUTINE FOR MAP USERS
*               WHO WISH TO REFER TO THE STD FN4 I/O PACKAGE.
*
*               THIS ROUTINE IS IDENTICAL IN FUNCTION
*               TO THE 7090 ROUTINE X022.  THIS 7044 VERSION IS
*               OFFERED COURTESY OF J D BABCOCK WITH THE BLESSINGS
*               G W ARMERDING WHO WAS RESPOSIBLE FOR THIS MESS
*               ON THE 7090.
*
*                 CALLS ARE---
*
*                       CAL   L (LOGICAL TPAE NO. IN DECREMENT)
*                       CALL    (ROUTINE ENTRY)
*                       PZE   FMT        (BCD ONLY)
*                       OP    A1,T1,N1
*                       OP    A2,T2,N2
*                       .     .  .  .
*                       .     .  .  .
*                       PZE   0            - LAST MUST BE ZERO.
*                             (RETURN)
*
*                 FMT IS LOCATION OF A STANDARD FN4 FORMAT STATEMENT
*                 AI,TI IS THE ADDR. OF FIRST DATA WORD (T=0,1,2)
*                 NI IS NUMBER OF WORDS (A(1)A-N+1)
*                 OP IS PZE FOR DIRECT ADDRESSING
*                 OP IS MZE FOR INDIRECT ADDRESSING
*
*               ROUTINE ENTRIES ARE---
*                       (1) RDATA --- BCD INPUT
*                       (2) IN ----  BCD INPUT FOR STD INPUT UNIT
*                             (CAL L NOT REQUIRED)
*                       (3) WDATA   BCD OUTPUT
*                       (4) OUT     BCD OUTPUT FOR STD OUTPUT UNIT
*                             (CAL L NOT REQUIRED)
*                       (5) PUNCH   BCD OUTPUT FOR STD PUNCH UNIT
*                             (CAL L NOT REQUIRED)
```

```
*                               (6) WBIN    BINARY OUTPUT
*                               (7) RBIN    BINARY INPUT
*
*
        ENTRY   WBIN
        ENTRY   RBIN
        ENTRY   IN
        ENTRY   RDATA
        ENTRY   OUT
        ENTRY   WDATA
        ENTRY   PUNCH
*
WBIN    AXT     0,0
        SXA     IR4,4
        AXT     3,4
        TRA     D1
*
RBIN    AXT     0,0
        SXA     IR4,4
        AXT     2,4
        TRA     D1
*
PUNCH   AXT     0,0                 ALLON CALLS OF
        CAL     =7                  TSL
        TRA     WDATA               OR
*                                   TSX VARIETY
IN      AXT     0,0
        CAL     =5                  STD INPUT UNIT
RDATA   AXT     0,0
        SXA     IR4,4
        AXT     0,4
        TRA     D1
*
OUT     AXT     0,0
        CAL     =6                  STD OUTPUT UNIT
WDATA   AXT     0,0
        SXA     IR4,4
        AXT     1,4
D1      SLW     DT                  SAVE LOGICAL TAPE NO.
        SXA     DR4,4
        AXT     7,4
        CAL*    ENTRY+7,4           TEST FOR TSX OR TSL ENTRY
        ANA     ADT
        TNZ     D7
        TIX     *-3,4,1
        MSP     ENTRY
DR4     AXT     **,4                WAS A TRUE TSX ENTRY
        TRA     D6
D7      PAC     0,4                 FIX
        TXI     *+1,4,-1            UP TRY TO SIMULATE
        CLA     IR4                 TSX ENTRY
        STA     TSLR4               SAVE PROG. IR4
        SXA     IR4,4
        LXA     DR4,4
        MSM     ENTRY
```

```
D6      CLA     SEL,4               TAPE
        STO     DSEL                SET UP TSH, STH
        CLA     END,4               AND FIL, RTN
        STO     DEND
        CLA     DV1,4
        STO     DT1
        CLA     DV2,4
        STO     DT2
        TXH     DBIN,4,1            TEST IF BCD OR BIN CALL
        CLA     HNL                 WAS BCD
        STO     CNVT                SET CONVERT CALL
        LXA     IR4,4
        CAL     1,4                SET UP FORMAT
        TXI     *+1,4,-1            BUMP FOR FIRST DATA CALL
        SXA     IR4,4
        ANA     ADT
        ORA     BCDFM
        TRA     GO
DBIN    CAL     NOP                 FOR BINARY NOP  2,4
        LDQ     BNL
        STQ     CNVT
GO      SLW     DFMT
        CAL     DT                     CALL TO SET UP
        TSX     UTVAR.,4            FILE NAME
*               GIVE INITIAL CALL
DSEL    ***     **                  TSX TSHIO,4 OR TSX STHIO,4
*                                   OR TSBIO.,STBIO.
DFILE   PZE     **                  FIL XX.
DFMT    PZE     **,0,**             FORMAT FROM CALL (BCD=FORMAT,,BIN=NO
*
*               NOW SET UP BASIC LOOP--
        LXA     IR4,4               PREPARE TO PULL OUT ARGUMENTS
D2      SXA     IR4,4
        CLA     1,4
        TZE     DEND                CHECK IF DONE
        PDC     0,4                 COUNT
        TXL     IR4,4,0             OUT IF NONE
        SXD     DTST,4              SET TEST FOR NO. OF WORDS
        TPL     D3                  CHECK IF INDIRECT
        ANA     ADT                 YES,, KILL DECREMENT  + PRFX.
        ORA     DCLA                SET UP CLA
        SLW     *+1
        ***     **                  GETS CLA A,T IF INDIRECT
D3      STA     D5                  SAVE A
        ANA     TAG                 PICK UP
        ORA     SXD4                PROGRAMMER INDEX REGISTER
        SLW     *+1                 AND PUT THE COMP. OF IT
        ***     **
        LDC     D4,4                INTO IR4
        SXD     D4,4
D5      AXT     **,4                COMPUTE ADDR OF A-T
D4      TXI     *+1,4,**
        SXA     DT1,4               SET ADDR.S OF PUTS
        SXA     DT2,4               AND GETS
        AXT     0,4                 BEGIN LOOP
```

```
DT1     ***                              NOP OR CLA DATA, T
CNVT    ***     **                       T=0(1)-N     (TSL HNLIO. OR BNLIO.
DT2     ***                              NOP OR STO DATA,T
*                                        NOP= AXT 0,0
        TXI     *+1,4,-1
DTST    TXH     DT1,4,**                 -N IN DECR
IR4     AXT     **,4
        TXI     D2,4,-1
*
DEND    ***     **                       FINAL EXIT, TSX FILIO.,4
*                                        OR TSX RTNIO.,4
*               RESTORE AXT 0,0 ENTRIES
        AXT     7,4
        PXA     0,0
        STA*    ENTRY+7,4
        TIX     *-1,4,1
        LXA     IR4,4
*
        MIT     ENTRY                    WAS IT TSX OR TSL
        TRA     DEND1                    WAS TSX OK-EXIT VIA 2,4
        TXI     *+1,4,-2                 WAS TSL, CALCULATE RETURN TRA
        PXA     0,4
        PAC     0,4                      ADDR
        SXA     DEND1-1,4
        LXA     TSLR4,4                  RESTORE PROG. IR4
        TRA     **                       RETURN TO MAIN PROGRAM
DEND1   TRA     2,4
*
BCDFM   MZE     **,,FMTSC.
BNL     TSL     BNLIO.
HNL     TSL     HNLIO.
DT      PZE                              LOGICAL TAPE
ADT     OCT     777777                   SAVE ADDR AND TAG
DCLA    CLA     **,0                     CLA ORDER
*               TABLE OF INITIAL CALLS
*
        TSX     STBIO.,4                 WBIN=IR4 OF 3
        TSX     TSBIO.,4                 RBIN=IR4 OF 2
        TSX     STHIO.,4                 WDATA = IR4 OF 1
SEL     TSX     TSHIO.,4                 RDATA = IR4 OF 0
*
        TSX     WLRIO.,4                 WBUN
        TSX     RLRIO.,4                 RBUN
        TSX     FILIO.,4                 WDATA
END     TSX     RTNIO.,4                 RDATA
*                         SET UP HNLIO. ,BNLIO. LOOPS
        CLA     **,4                     3
        AXT     0,0                      2
        CLA     **,4                     1
DV1     AXT     0,0                      0
*
        AXT     0,0                      3
        STO     **,4                     2
        AXT     0,0                      1
DV2     STO     **,4                     0
```

```
*
*
ENTRY  PZE    PUNCH
       PZE    WDATA
       PZE    OUT
       PZE    RDATA
       PZE    IN
       PZE    WBIN
       PZE    RBIN
*
TSLR4  PZE    **
TAG    OCT    000000700000
SXD4   SXD    D4,0
NOP    EQU    OV1
*
       EXTERN UTVAR.
       EXTERN TSBIO.
       EXTERN STBIO.
       EXTERN BNLIO.
       EXTERN TSHIO.
       EXTERN STHIO.
       EXTERN HNLIO.
       EXTERN FMTSC.
       EXTERN RTNIO.
       EXTERN FILIO.
       EXTERN WLRIO.
       EXTERN RLRIO.
       END


$IBMAP UTV
       ENTRY  UTVAR.
       EXTERN ERLOC.
UTVAR. SXA    UTVX,4                  SAVE RETURN INDEX
       LAC    UTVX,4
       SXA    ERLOC.,4
       LXA    UTVX,4
       LAS    NFILES                  STOP IF LOGICAL TAPE NUMBER EXCEEDS
       TRA    USTOP                      NUMBER OF FILES IN TABLE.
       NOP
       PAC    ,4
       CLA    IOU,4                   PICKUP ADDRESS OF FCB POINTER
       PAX    ,4
       TXL    USTOP-2,4,0             STOP IF UNIT IS UNDEFINED
UTVX   AXT    **,4                    RESTORE RETURN INDEX
       STO    2,4                     SET LOCATION OF FCB
       TRA    1,4                     RETURN TO MAIN PROGRAM
       LXA    UTVX,4
       CLA*   -1,4                    RESTORE UNIT DESIGNATION
USTOP  TSL    FEXEM.                  ERROR, ILLEGAL UNIT REQUESTED.
       PZE    EXIT,,32                NO OPTIONAL RETURN
*INPUT-OUTPUT LOGICAL UNIT TABLE
*ADDITIONS OR DELETIONS SHOULD BE MADE BETWEEN IOU AND NFILES
IOU    PZE    FIL00.
```

```
        PZE     FIL01.
        PZE     FIL02.
        PZE     FIL03.
        PZE     FIL04.
        PZE     FIL05.
        PZE     FIL06.
        PZE     FIL07.
        PZE     FIL08.                                              ******
        PZE     FIL09.                                              ******
        PZE     FIL10.                                              ******
        PZE     FIL11.                                              ******
        PZE     FIL12.                                              ******
        PZE     FIL13.                                              ******
        PZE     FIL14.                                              ******
        PZE     FIL15.                                              ******
NFILES  PZE     *-IOU-1
        EXTERN  FIL00.                                              ******
        EXTERN  FIL01.                                              ******
        EXTERN  FIL02.                                              ******
        EXTERN  FIL03.                                              ******
        EXTERN  FIL04.                                              ******
        EXTERN  FIL05.                                              ******
        EXTERN  FIL06.                                              ******
        EXTERN  FIL07.                                              ******
        EXTERN  FIL08.                                              ******
        EXTERN  FIL09.                                              ******
        EXTERN  FIL10.                                              ******
        EXTERN  FIL11.                                              ******
        EXTERN  FIL12.                                              ******
        EXTERN  FIL13.                                              ******
        EXTERN  FIL14.                                              ******
        EXTERN  FIL15.                                              ******
        EXTERN  FEXEM.,EXIT
        END
```

REFERENCES

1. Liechenstein, M. I., *Orthonormal Bases for Exponential and Rational Function Approximations of Network and Signal Characteristics,* The RAND Corporation, RM-5177-PR, January 1967.

2. Wiener, N., and R. E. A. C. Paley, "Fourier Transforms in the Complex Domain," *Am. Math. Soc. Colloq. Pubs.,* Vol. 19, New York, 1934, p. 26.

3. Sokolnikoff, I. S., and R. M. Redheffer, *Mathematics of Physics and Modern Engineering,* McGraw-Hill Book Co., Inc., New York, 1958, pp. 118-120.

4. Wiener, N., *The Fourier Integral and Certain of Its Applications,* Cambridge University Press, Cambridge, England, 1933, pp. 60-70.

5. Davis, P. J., *Interpolation and Approximation,* Blaisdell Publishing Co., New York, 1963, pp. 165-166.

6. Whittaker, E. T., and G. N. Watson, *A Course of Modern Analysis,* Cambridge University Press, Cambridge, England, 1962, pp. 115-116.

7. Kaplan, W., *Operational Methods for Linear Systems,* Addison-Wesley Publishing Co., Inc., Palo Alto, California, 1962, pp. 125-131.

8. Abramowitz, M., and I. A. Stegun (eds.), *Handbook of Mathematical Functions,* National Bureau of Standards Applied Mathematics Series, AMS 55, Washington, D. C., 1964, pp. 824-825.

9. Burnside, W. S., and A. W. Panton, *The Theory of Equations,* Vol. 1, Dover Publications, Inc., New York, 1960, pp. 22-23, 259-261.

10. Hamming, R. W., *Numerical Methods for Scientists and Engineers,* McGraw-Hill Book Co., Inc., New York, 1962, p. 8.

11. Stewart, B. M., *Theory of Numbers,* The Macmillan Company, New York, 1952, p. 19.

12. Hadley, G., *Nonlinear and Dynamic Programming,* Addison-Wesley Publishing Co., Inc., Palo Alto, California, 1964, pp. 53-58.

13. Leitmann, G. (ed.), *Optimization Techniques,* Academic Press, New York, 1962, p. 6.

14. Bellman, R., R. E. Kalaba, and J. Lockett, *Numerical Inversion of the Laplace Transform,* American Elsevier Publishing Co., Inc., New York, 1966, Chap. 1.

15. Lanczos, C., *Applied Analysis,* Prentice Hall, Inc., Englewood Cliffs, New Jersey, 1961, pp. 280-303.

16. Macon, N., and A. Spitzbart, "Inverses of Vandermonde Matrices," *Am. Math. Monthly*, Vol. 65, February 1958, pp. 95-100.

17. Laning, J. H., and R. H. Battin, *Random Processes in Automatic Control*, McGraw-Hill Book Co., Inc., New York, 1956, pp. 422-423.

18. Klinger, A., "The Vandermonde Matrix," *Am. Math. Monthly*, Vol. 74, No. 5, May 1967, pp. 571-574.

19. Liechenstein, M. I., *Evaluation of the Determinental Polynomial of a Matrix of Polynomial Elements and an Application to AMTI Radar*, The RAND Corporation, RM-483 -PR, January 1966, pp. 11-15, 43-44, 83-120.

20. Hildebrand, F. B., *Introduction to Numerical Analysis*, McGraw-Hill Book Co., Inc., New York, 1956, pp. 378-382.

21. Bellman, R., *On the Separation of Exponentials*, The RAND Corporation, P-1846, December 1959.

22. Gardner, D. G., J. C. Gardner, G. Laush, and W. W. Meinke, "Method for the Analysis of Multicomponent Exponential Decay Curves," *J. Chem. Phys.*, Vol. 31, No. 4, October 1959, pp. 978-986.

23. Wynn, P., "A Note on the Fitting of Certain Types of Experimental Data," *Statistica Neerlandica*, Vol. 16, 1962, pp. 143-150.

24. Cornell, R. G., *A New Estimation Procedure for Linear Combinations of Exponentials*, Oak Ridge National Laboratory, Report ORNL-2120, September 1956.

25. Slepian, D., "Estimation of Signal Parameters in the Presence of Noise," *IRE Trans. Info. Theory*, Vol. IT-3, March 1954, pp. 68-89.

26. Bellman, R. E., and R. E. Kalaba, *Quasilinearization and Nonlinear Boundary-Value Problems*, American Elsevier Publishing Co., Inc., New York, 1965, pp. 139-150.

27. Hudson, G. E., "Trend Analysis of Physical Data," *Am. J. Phys.*, Vol. 21, No. 5, May 1953, pp. 362-367.

28. Litman, S., and W. H. Huggins, "Growing Exponentials as a Probing Signal for System Identification," *Proc. IEEE*, Vol. 51, No. 6, June 1963, pp. 917-923.

29. Worseley, B. H., "Analysis of Decay-Type Data," *Comm. ACM*, Vol. 7, No. 1, January 1964, pp. 39-44.

30. Liechenstein, M. I., "Rational Function Approximations for Arbitrary Envelope Delay," Ph.D. dissertation, Yale University, June 1965, Chap. 2.

31. Prony, R., "Essai expérimental et analytique," *Journal de l'école polytechnique*, Paris, Vol. 1, No. 2, 1795, pp. 24-35.

32. Householder, A. S., *On Prony's Method of Fitting Exponential Decay Curves and Multiple-Hit Survival Curves*, Oak Ridge National Laboratory, Report ORNL-455, September 1949.

33. Kautz, W. H., "Transient Synthesis in the Time Domain," *Trans. IRE*, Vol. CT-1, No. 3, September 1964, pp. 29-39.

34. Mathers, G. W. C., *The Synthesis of Lumped-Element Circuits for Optimum Transient Response*, Electronics Research Laboratory, Stanford University, Tech. Rept. 38, November 1951.

35. Wiener, N., *Extrapolation, Interpolation, and Smoothing of Stationary Time Series*, John Wiley and Sons, Inc., New York, 1960, Chaps. 2,3.

36. Bendat, J. S., *Principles and Applications of Random Noise Theory*, John Wiley and Sons, Inc., New York, 1958, Section 7.3.

37. Harman, W. W., *Principles of the Statistical Theory of Communication*, McGraw-Hill Book Co., Inc., New York, 1963, pp. 43-48.

38. Brown, W. S., "A Language and System for Symbolic Algebra," in *System Analysis by Digital Computer*, F. F. Kuo and J. F. Kaiser (eds.), John Wiley and Sons, Inc., New York, 1966, pp. 349-373.

DOCUMENT CONTROL DATA

1. ORIGINATING ACTIVITY THE RAND CORPORATION	2a. REPORT SECURITY CLASSIFICATION UNCLASSIFIED 2b. GROUP
3. REPORT TITLE TWO-PARAMETER EXPONENTIAL AND RATIONAL FUNCTIONS FOR LEAST-SQUARE APPROXIMATIONS	
4. AUTHOR(S) (Last name, first name, initial) Liechenstein, M. I.	
5. REPORT DATE September 1968	6a. TOTAL No. OF PAGES 142 / 6b. No. OF REFS. 38
7. CONTRACT OR GRANT No. F44620-67-C-0045	8. ORIGINATOR'S REPORT No. RM-5707-PR
9a. AVAILABILITY/ LIMITATION NOTICES DDC-1	9b. SPONSORING AGENCY United States Air Force Project Rand

10. ABSTRACT

A computationally efficient scheme for approximating system characteristics by sums of exponentials or by rational functions. Results are applicable to a broad class of system optimizations, such as those in network synthesis and radar filter design, and to the understanding of cross-correlation measurements, radioactive decay, gas absorption, and mass spectrograph and ultracentrifuge analysis curves. Two sets of two-parameter orthonormal elements are derived: one set constitutes a basis for exponential approximation and the other a basis for rational function approximation. The closure properties of the two orthonormal bases are examined and new expressions are developed for efficiently determining the orthonormal elements of each basis. Several relations are then deduced that connect important properties of each basis. Useful identities and numerical techniques involving the basis coefficients are derived that obviate storage of either the form or selected values of the orthonormal approximants. Finally, several numerical examples illustrate algorithms for both exponential and rational function approximation. Computer programs in ALTRAN and in the more efficient FORTRAN IV are appended.

11. KEY WORDS

Curve fitting
Numerical methods and processes
Radar
Radio
Communication systems
Signal processing